

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Глухов В.С., Костик А.Т.

**ДОСЛІДЖЕННЯ І ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ
СИСТЕМ ТА МЕРЕЖ**

Навчальний посібник

Львів
Видавництво "Магнолія 2006"
2024

УДК 004.3

Автори : Глухов В.С., д.т.н., професор, професор кафедри ЕОМ,
Костик А.Т., асистент кафедри ЕОМ.

Рецензенти : Оліярник Б.О., д.т.н., професор,
Головний конструктор Державного підприємства «Львівський
держваний завод «ЛОРТА»»,
Адольф Лукенюк., к.т.н., директор Львівського центру
Інституту космічних досліджень НАН та ДКА України,
Бочкарьов О.Ю., к.т.н., доцент кафедри електронних
обчислювальних машин, Національний університет
“Львівська політехніка”

Рекомендовано Науково-методичною радою
Національного університету “Львівська політехніка”
(протокол № 1 від 21 лютого 2023 року)

Навчальний посібник з дисципліни «Дослідження та проектування
комп'ютерних систем та мереж» : для студентів другого (магістерського) рівня
вищої освіти спеціальності 123 ”Комп'ютерна інженерія”. Навчальний
посібник / Автори : В.С. Глухов, А.Т. Костик - Львів: Видавництво "Магнолія
2006", 2024. – 253 с.

Навчальний посібник призначений для студентів спеціальності
“Комп'ютерна інженерія” галузі знань “Інформаційні технології”.

ISBN 978-617-574-265-5

УДК 004.3

© Глухов В. С., Костик А.Т., 2024
© Видавництво "Магнолія 2006", 2024

ЗМІСТ

Вступ.....	5
1. Проектування інтегральних схем.....	7
1.1. Процес проектування в середовищі Active-hdl	7
1.1.1. Опис системи Active-hdl.....	7
1.1.2. Створення і відлагодження проекту.....	9
1.1.3. Логічні елементи та їх класифікація	10
1.2. Лабораторна робота № 1. Використання вбудованих бібліотечних елементів у комп'ютерних системах.....	14
1.2.1. Мета та завдання лабораторної роботи.....	14
1.2.2. Порядок виконання лабораторної роботи	14
1.2.3. Генератор ядер	23
1.2.4. Результати роботи.....	28
1.3. Лабораторна робота № 2. Проектування макроелементів	29
1.3.1. Мета та завдання лабораторної роботи.....	29
1.3.2. Порядок виконання роботи	29
1.3.3. Результати роботи.....	39
Контрольні питання.....	39
2. Перетворювач двійково-десяткових кодів десятичних цифр	41
2.1. Пристрої перетворення	41
2.1.1. Основа системи числення	41
2.1.2. Шифратор та дешифратор.....	42
2.1.3. ПЗП та ОЗП.....	43
2.1.4. FIFO та LIFO.....	45
2.2. Лабораторна робота № 3. Проектування та дослідження перетворювачів кодів	45
2.2.1. Мета та завдання лабораторної роботи.....	45
2.2.2. Порядок виконання лабораторної роботи	47
2.2.3. Результати роботи.....	56
2.3. Лабораторна робота № 4. Проектування та дослідження регістрових схем	57
2.3.1. Мета та завдання лабораторної роботи.....	57
2.3.2. Порядок виконання роботи	57
2.3.3. Результати роботи.....	66
2.4. Лабораторна робота № 5. Проектування та дослідження вузлів пам'яті вбудованих систем	67
2.4.1. Мета та завдання лабораторної роботи.....	67
2.4.2. Порядок виконання роботи	69
2.4.3. Результати роботи.....	74
2.5. Лабораторна робота № 6. Проектування та дослідження регістрових файлів вбудованих систем	75
2.5.1. Мета та завдання лабораторної роботи.....	75
2.5.2. Порядок виконання роботи	76
2.5.3. Результати роботи.....	83
2.6. Лабораторна робота № 7. Проектування та дослідження операційних пристроїв вбудованих систем	84
2.6.1. Мета та завдання лабораторної роботи.....	84
2.6.2. Порядок виконання роботи	86

2.6.3. Результати роботи.....	90
2.7. Лабораторна робота № 8. Проектування та дослідження вбудованих процесорів...	90
2.7.1. Мета та завдання лабораторної роботи.....	90
2.7.2. Порядок виконання роботи	92
2.7.3. Результати роботи.....	101
Контрольні питання.....	101
3. Паралельні помножувачі.....	102
3.1. Лабораторна робота № 9. Проектування та дослідження паралельних помножувачів для розширених полів Галуа $GF(3^4)$	102
3.1.1. Мета та завдання лабораторної роботи.....	102
3.1.2. Теоретичні відомості	102
3.1.3. Порядок виконання роботи	107
3.1.4. Результати роботи.....	107
3.2. Лабораторна робота № 10. Синтез високого рівня (High-Level Synthesis - HLS)...	108
3.2.1. Мета та завдання лабораторної роботи.....	108
3.2.3. Результати роботи.....	116
Контрольні питання.....	116
Тести та відповіді	117
Глосарій	248
Література	251

Навчальне видання

ДОСЛІДЖЕННЯ І ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ

Навчальний посібник

Підп. до друку 24.09.2023 р.
Формат 70x100/16. Папір друк. № 2. Гарнітура Times New Roman .
Умовн. друк. арк. 20,56. Наклад 50 прим.

ПП «Магнолія 2006»
м. Львів-53, 79053, Україна, тел. +38 (050)370-19-57
e-mail: magnol06@ukr.net
<https://magnolia.lviv.ua>

Свідоцтво про внесення суб'єкта видавничої справи до
Державного реєстру видавців, виготівників
і розповсюджувачів видавничої продукції:
серія ДК № 2534 від 21.06.2006 року, видане Державним комітетом
інформаційної політики, телебачення
та радіомовлення України

Надруковано у друкарні видавця Марченко Т. В.

1. Проектування інтегральних схем

1.1. Процес проектування в середовищі Active-hdl

1.1.1. Опис системи Active-hdl

Система Active-HDL – продукт, орієнтований на максимальну гнучкість розробки складних проектів з урахуванням будь-яких типів сучасних ПЛІС різних компаній-виробників. Важливо те, що окрім розвинених вбудованих засобів моделювання та налагодження Active-HDL також надає розробникам єдине середовище для побудови власних маршрутів проектування ПЛІС залежно від характеру розв'язуваних завдань та програмних засобів.

Центральне місце у структурі системи Active-HDL займає блок управління маршрутом проектування (Рис.1).

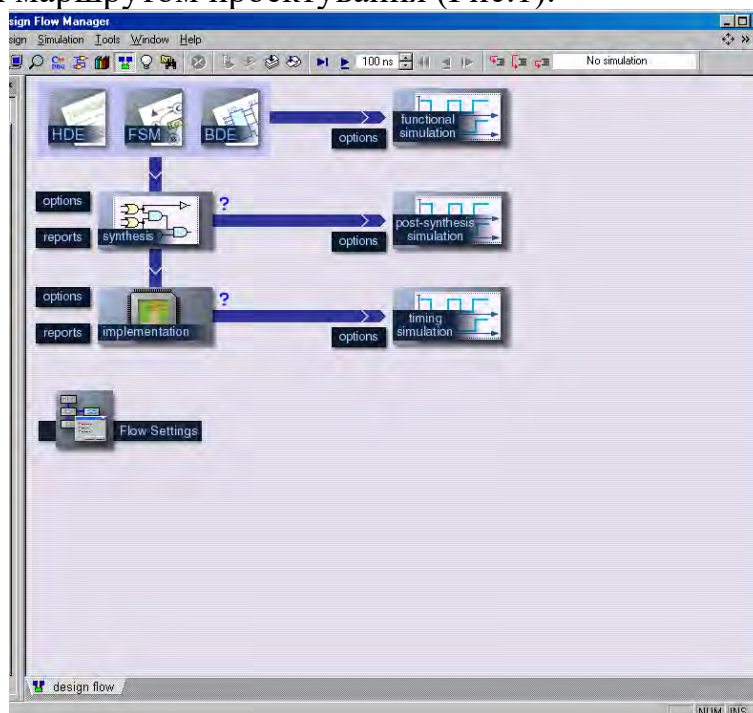


Рис.1.1. Структура системи Active-HDL

У блоці управління розробник може визначити набір засобів, що використовуються на кожному з етапів проектування ПЛІС (моделювання, синтез, проектування топології, спеціальні засоби), які надалі працюватимуть у рамках єдиного маршруту та середовища проектування системи Active-HDL. У системі в будь-яких комбінаціях можуть бути використані всі доступні на сьогоднішній день кошти виробників ПЛІС та більшість популярних продуктів компаній – виробників САПР. Склад коштів, які можна вбудовані у маршрут проектування, постійно розширюється. Інтеграція різних продуктів у єдиному середовищі дозволяє повністю контролювати процес проектування по всьому його протязі: від специфікації до фізичної реалізації. Розробник може налаштувати маршрут проектування саме на той склад програмних засобів,

Окрім власних технологічно незалежних бібліотек Active-HDL включає повні бібліотеки всіх виробників ПЛІС із прекомпільованими та готовими до

використання компонентами. Коли маршрут проектування вже налаштований, всю послідовність операцій із проектування ПЛІС можна проводити із середовища блоку управління. Для запуску будь-якої підсистеми та передачі їй усієї необхідної проектної інформації просто вибирається кнопка відповідного компонента маршруту. Можна також встановити режим виконання (графічний або фоновий).

Мультипроектне робоче середовищедопускає супровід відразу кількох незалежних проектів, кожному з яких може бути передбачена власна конфігурація. Якщо це модулі однієї системи, вони згодом поєднуються у проекті верхнього рівня. Засоби управління завданнями дозволяють організувати роботу в мережі та оптимізувати обчислювальний процес. Виконання завдань моделювання, логічного синтезу, топологічної реалізації, що потребують значних витрат часу та обчислювальних ресурсів, може бути переведено на віддалені сервери, вивільнивши комп'ютер розробника для інтерактивних завдань. Центральний планувальник управляє чергою завдань, отриманих від усіх користувачів, та розподілом вільних обчислювальних ресурсів. Після виконання завдання планувальник сповіщає користувача про те, що він може завантажити результати на свій локальний комп'ютер для аналізу. Підтримується такий спосіб організації багатомашинної системи, як ферма серверів із керуванням пріоритетами завдань у чергах. Для організації групової розробки великих проектів Active-HDL надає також інтерфейси до п'ятнадцяти найпопулярніших систем управління версіями (Revision Control System – RCS). Після реєстрації будь-якої системи RCS у Active-HDL все керування версіями виконується безпосередньо у середовищі Active-HDL.

Робота з проектами у середовищі блоку управління передбачає наступний набір операцій:

- додавання, видалення, перегляд, модифікація файлів ресурсів проекту;
- перегляд вмісту робочої бібліотеки, бібліотеки результатів логічного синтезу та бібліотеки тимчасових параметрів поточного проекту;
- перегляд структури проектного модуля, що моделюється;
- перегляд заданих об'єктів усередині проектного модуля, що моделюється.

Для роботи з бібліотеками передбачені операції:

- підключення, вимкнення, видалення бібліотек;
- редагування логічних імен бібліотек;
- стиснення та видалення вмісту бібліотек;
- перегляд вмісту бібліотек;
- перегляд бібліотечних модулів;
- видалення бібліотечних модулів;
- пошук модулів у бібліотеках.

1.1.2. Створення і відлагодження проекту

Система Active-HDL підтримує різноманітні засоби створення та редагування проекту ПЛІС, які включають: текстовий редактор апаратних специфікацій на базі мов VHDL, Verilog, SystemC, а також C/C++/Handel-C та Matlab (HDE); графічні редактори діаграм автоматів кінцевих станів (FSM) та блок-діаграм (BDE). Текстовий редактор розпізнає та виділяє синтаксичні конструкції, в ньому передбачені функції автозаповнення, а також підтримка повних наборів шаблонів мов, що використовуються відповідно до стандартів.

Підтримується опис схеми у форматі EDIF, імпорт схемних символів, ієрархічна візуалізація опису проекту, генерація графічного представлення у вигляді блок-діаграм та діаграм автоматів кінцевих станів за описом HDL. Генератор IP-блоків дозволяє автоматично створювати синтезовані моделі для різних арифметичних функцій, послідовної логіки, блоків пам'яті, фільтрів, конверторів кодів, елементів системи тестування, а також деяких спеціальних промислових і комунікаційних додатків.

Моделювання змішаних проектів, що включають описи мов VHDL, Verilog, списки ланцюгів у форматі EDIF і моделі SystemC, підтримується на рівні самого ядра моделювання Active-HDL. Вбудовані інтерфейси моделювання дозволяють додатково використовувати поведінкові моделі мовами C/C++/Handel-C, функціональні моделі SWIFT, моделі пам'яті Denali, а також вбудовані та M-функції системи Matlab.

Засоби крос-налагодження сигналів дозволяють не тільки відстежувати правильність часових діаграм сигналу в потрібних точках проекту, але й забезпечують перехід із редактора діаграм до відповідного елемента вихідного опису (HDL-текст, блок-діаграми). Є також можливість організації у заданих внутрішніх точках додаткового аналізу подій для автоматичного виявлення помилкових та сумнівних ситуацій. Для великих проектів і за можливості заздалегідь провести моделювання зручне налагодження у відкладеному режимі. У такому режимі розробник використовує у процесі налагодження заздалегідь підготовлені дані, не витрачаючи часу очікування відгуку системи моделювання.

У процесі моделювання також можна генерувати графічне представлення потоків даних моделі, яке дозволяє досліджувати зв'язність моделі і провести аналіз обміну даними між будь-якими її об'єктами (компонентами, регістрами, сигналами, операторами). Візуалізація шляхів розповсюдження сигналу за допомогою діаграм потоків даних дає змогу легше визначити потенційні джерела помилок на внутрішніх сигналах.

Система Active-HDL підтримує стандарти VHDL, Verilog, VITAL та SDF. Підтримуються також інтерфейси Tcl/Tk, PERL, SWIFT, PLI/VPI, VHPI та CHPI. Система функціонує в операційному середовищі Microsoft Windows. Моделювання можна також проводити на віддалених комп'ютерах, що працюють у середовищі Linux або Unix (за рахунок використання власної системи моделювання компанії Aldec – Riviera-PRO).

1.1.3. Логічні елементи та їх класифікація

Логічний елемент – це електронний пристрій, що реалізує одну з логічних операцій. Логічні елементи являють собою електронні пристрої, у яких оброблювана інформація закодована у вигляді двійкових чисел. Термін «логічні» прийшов в електроніку з алгебри логіки, що оперує зі змінними величинами і їхніми функціями, що можуть приймати тільки два значення: «істинно» чи «хибно». Для позначення чи істинності хибності висловлень використовують відповідно символи 1 чи 0. Кожна логічна перемінна може приймати тільки одне значення: 1 чи 0. Ці двійкові змінні і функції від них називаються логічними змінними і логічними функціями.

Для одного аргументу маємо $2^1 = 2$ набори і $2^2 = 4$ чотири логічні функції від однієї змінної.

Фактична кількість ФАЛ1 дорівнює $A_1 = 2^{2^1} - A_0 = 4 - 2 = 2$. У таблиці (таблиця 1.1) відмічено, які з ФАЛ1 ($f_0(a)$ і $f_3(a)$) фактично є функціями алгебри логіки нуля змінних (ФАЛ0).

Таблиця 1.1. Функції алгебри логіки однієї змінної

a	$f_0(a)$	$f_1(a)$	$f_2(a)$	$f_3(a)$
0	0	0	1	1
1	0	1	0	1
Фактично є	ФАЛ0			ФАЛ0

Функція $f_1(a)$, що повторює абсолютне значення логічної змінної, - функція тотожності.

Варіанти аналітичного запису $f_1(a) = a$; $f_1(a) \equiv a$.

Наведена вище формула читається так: f_1 дорівнює a.

Інші назви для функція тотожності – функція повторення.

Умовні графічні позначення (символ) пристрою (елемента цифрової техніки), який реалізує функцію тотожності (елемента повторювача) наведені на рисунку 1.2., де (і далі) позначено:

а – умовне графічне позначення згідно з стандартом ГОСТ 2.743-91;

б - умовне графічне позначення згідно з міжнародним стандартом.

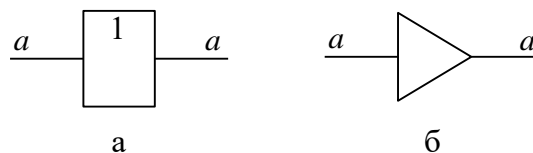


Рис. 1.2. Повторювач

Символи елемента, який реалізує функцію інверсії (інвертора) наведено на рисунку 1.3.

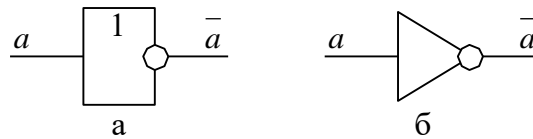


Рис. 1.3. Інвертор (елемент НЕ)

Диз'юнкція (від латинського *disjunctio* - роз'єднання), логічне додавання або функція АБО (ИЛИ, OR) - це функція $f_7(a,b)$, що істинна тоді, коли істинна хоча б одна з її змінних (таблиця 7.).

Таблиця 1.2. Таблиця істинності диз'юнкції (АБО)

Змінні (аргументи)		Функція
a	b	$a \vee b$
0	0	0
0	1	1
1	0	1
1	1	1

Варіанти аналітичного запису цієї функції: $f_7(a,b) = a \vee b$ і зараз вже дуже рідко $a + b$.

Це читається в такий спосіб: f_7 дорівнює a або b .

Це значить, що функція істинна, тобто дорівнює одиниці, коли аргумент $a = 1$, тобто істинний, або аргумент $b = 1$, або ж обидва аргументи істинні одночасно.

Символи елемента, який реалізує функцію диз'юнкції (диз'юнктора, елемента АБО) наведено на рисунку 1.4.

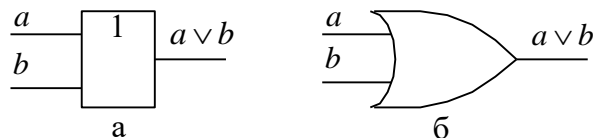


Рис. 1.4. Диз'юнктор (елемент АБО)

Кон'юнкція (від латинського *conjunctio* – сполучник, зв'язок), логічне множення або функція І (И, AND) - це функція $f_1(a,b)$, що істинна тоді, коли всі її змінні одночасно істинні (таблиця 1.3).

Таблиця 1.3. Таблиця істинності кон'юнкції (І)

Змінні (аргументи)		Функція
a	b	$a \& b$
0	0	0
0	1	0
1	0	0
1	1	1

Варіанти аналітичного запису цієї функції: .

Це читається в такий спосіб: f_1 дорівнює ab (можна читати і так: f_1 дорівнює a і b).

Це значить, що функція істинна, тобто дорівнює одиниці, коли обидва аргументи одночасно істинні, тобто дорівнюють одиниці.

Символи елемента, який реалізує функцію кон'юнкції (кон'юнктора, елемента І) наведені на рисунку 1.5.

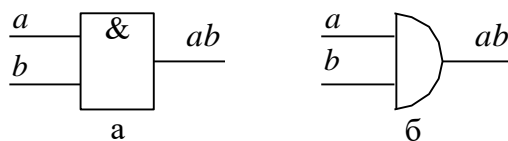


Рис. 1.5. Кон'юнктор (елемент І)

Функція (штрих) Шеффера або функція І-НЕ (NOT AND, NAND, И-НЕ) - це функція $f_{14}(a,b)$, що хибна тоді, коли всі змінні істинні (таблиця 1.4).

Таблиця 1.4. Таблиця істинності функції Шеффера (І-НЕ)

Змінні (аргументи)		Функція
a	b	$\overline{a \& b}$
0	0	1
0	1	1
1	0	1
1	1	0

Умовна позначка цієї функції:

$$f_{14}(a,b) = \overline{a \& b} = \overline{ab} = \overline{a \cdot b} = \overline{a \wedge b} = a / b.$$

Це читається в такий спосіб: f_{14} дорівнює не ab .

Ця функція хибна, тобто дорівнює 0, коли обидва аргументи одночасно істинні, тобто дорівнюють одиниці, і функція істинна, тобто дорівнює одиниці, коли або обидва аргументи одночасно хибні, або ж хоча б один з них хибний.

Символи елемента, який реалізує функцію Шеффера (І-НЕ) наведені на рисунку 1.6.

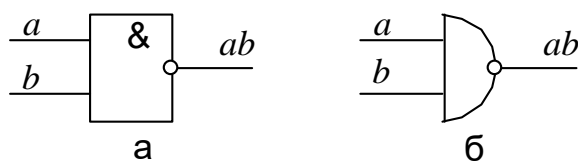


Рис. 1.6. Елемент Шеффера (І-НЕ)

Таблиця 1.5. Таблиця істинності функції Пірса (АБО-НЕ)

Змінні (аргументи)		Функція
a	b	$\overline{a \vee b}$
0	0	1
0	1	0
1	0	0
1	1	0

Символи елемента, який реалізує функцію Пірса (АБО-НЕ) наведені на рисунку 1.7.

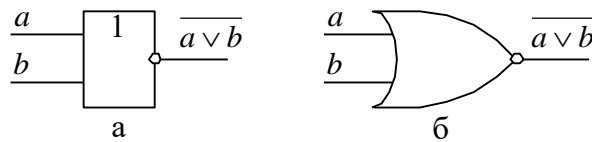


Рис. 1.7. Елемент Пірса (АБО-НЕ)

Виключне АБО (XOR) - це функція $f_6(a,b)$, що позначається знаком \oplus . Ця операція реалізує функцію нерівнозначності, тобто фактично реалізується процедура додавання за модулем 2 (таблиця 1.6).

Таблиця 1.6. Таблиця істинності функції виключне АБО

Змінні (аргументи)		Функція
a	b	$a \oplus b$
0	0	0
0	1	1
1	0	1
1	1	0

Ця функція хибна, тобто дорівнює 0, коли її аргументи мають однакові значення, і функція істинна, тобто дорівнює одиниці, коли аргументи мають різні значення (істинне значення має виключно один з аргументів).

Символи елемента, який реалізує функцію нерівнозначності (виключне АБО) наведені на рисунку 1.8.

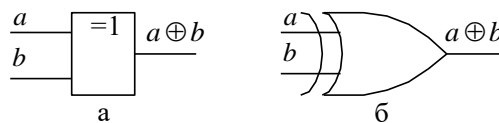


Рис. 1.8. Елемент виключне АБО

Таблиця 1.7. Таблиця істинності функції рівнозначності

Змінні (аргументи)		Функція
a	b	$a \equiv b$
0	0	1
0	1	0
1	0	0
1	1	1

Це читається в такий спосіб: f_9 істина, коли a дорівнює b .

Ця функція хибна, тобто дорівнює 0, коли її аргументи мають різні значення, і функція істинна, тобто дорівнює одиниці, коли аргументи мають однакові значення.

Символи елемента, який реалізує функцію рівнозначності (еквівалентності) наведені на рисунку 1.9.

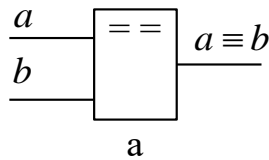


Рис. 1.9. Елемент рівнозначності (еквівалентності)

1.2. Лабораторна робота № 1. Використання вбудованих бібліотечних елементів у комп'ютерних системах

1.2.1. Мета та завдання лабораторної роботи

Засвоєння навичок роботи у середовищі Active-HDL, засвоєння методів створення бібліотечних елементів вузлів вбудованих комп'ютерних систем.

Задати бібліотечний елемент можна:

- його внутрішньою схемою;
- за допомогою генератора ядер;
- HDL-описом.

1.2.2. Порядок виконання лабораторної роботи

1. Запустити на виконання програму Active-HDL. Повинне з'явитися робоче вікно програми, як показано на рис. 1.10.

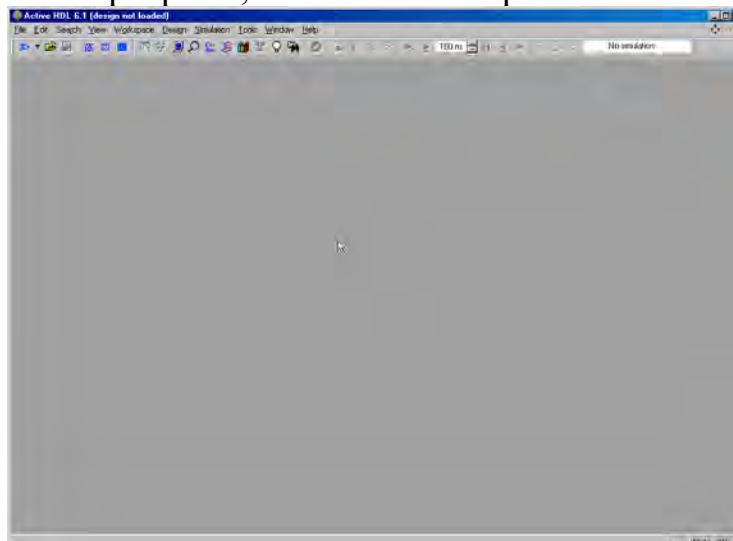


Рис. 1.10. Робоче вікно програми Active-HDL.

2. Почати створення нового проекту. Для цього слід виконати підпункт меню “File->New->Workspace...” (рис. 1.11).

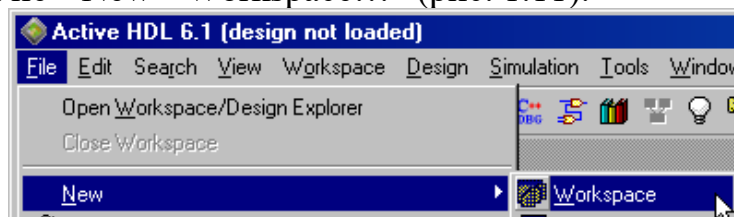


Рис. 1.11. Створення нового проекту.

3. У вікні нового проекту необхідно дати проекту ім'я, яке збігається з прізвищем. Для цього у полі “Type the workspace name” написати латиницею своє прізвище і натиснути клавішу “OK” (рис. 1.12).

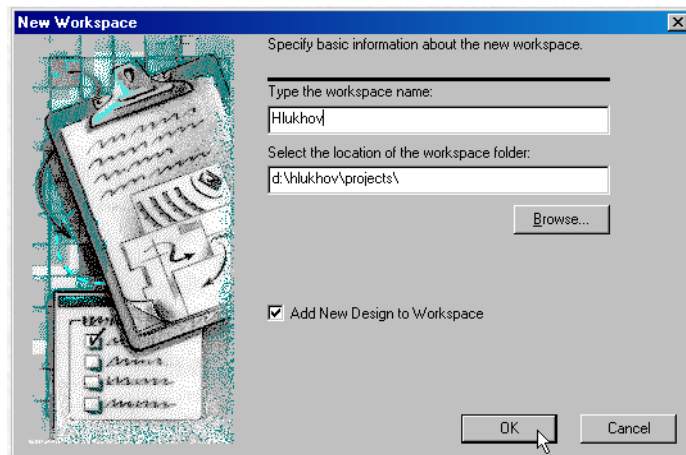


Рис. 1.12. Ім'я нового проекту.

4. Поставити позначку на “Create an empty design”, яка вказує, що створюваний проект повинний бути порожнім, тобто не містити ніяких файлів та натиснути клавішу “Next” (рис. 1.13).

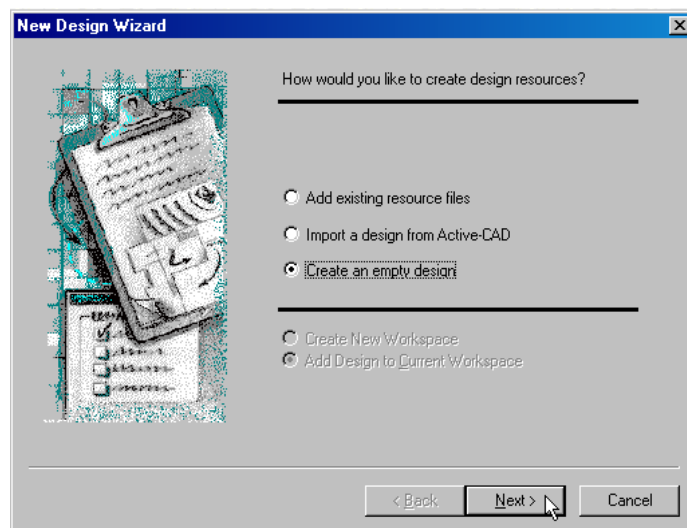


Рис. 1.13. Створення порожнього проекту.

5. Задати, які програмні засоби будуть використовувати даний проект та натиснути клавішу “Next” (рис. 1.14 – 1.15).

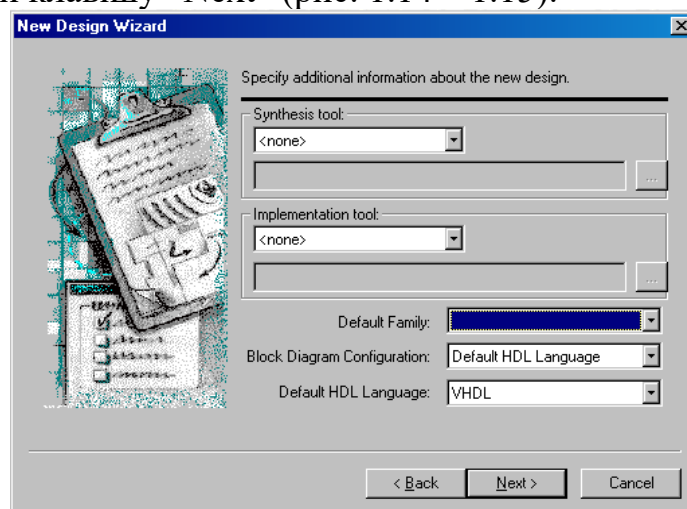


Рис. 1.14. Вибір програмних засобів.

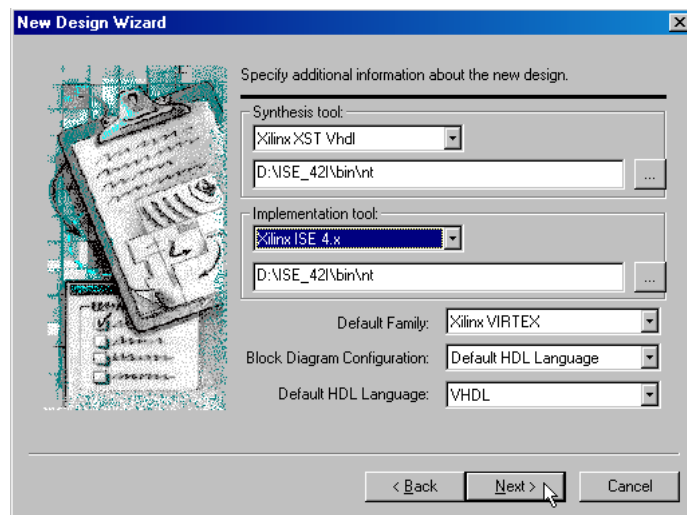


Рис. 1.15. Індивідуальне налаштування програмних засобів.

6. Після вибору програмних засобів задати ім'я першої лабораторної роботи та натиснути клавішу "Next" (рис. 1.16).

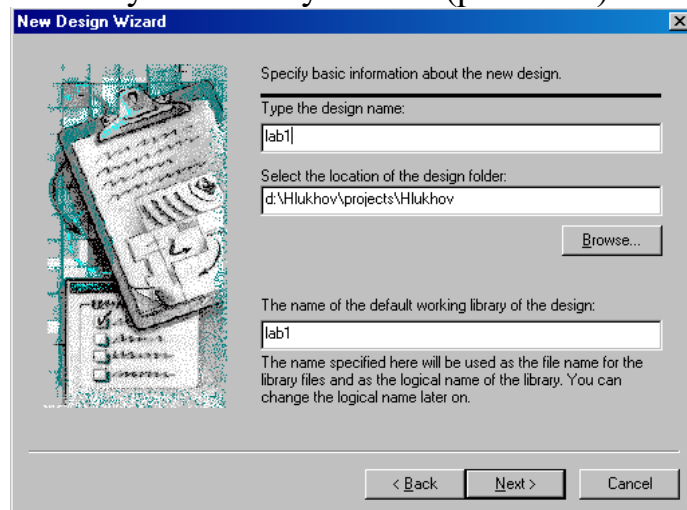


Рис. 1.16. Ім'я лабораторної роботи.

7. Після вибору імені проекту, натиснути клавішу "Finish", що дозволить завершити створення нового проекту (рис. 1.17).

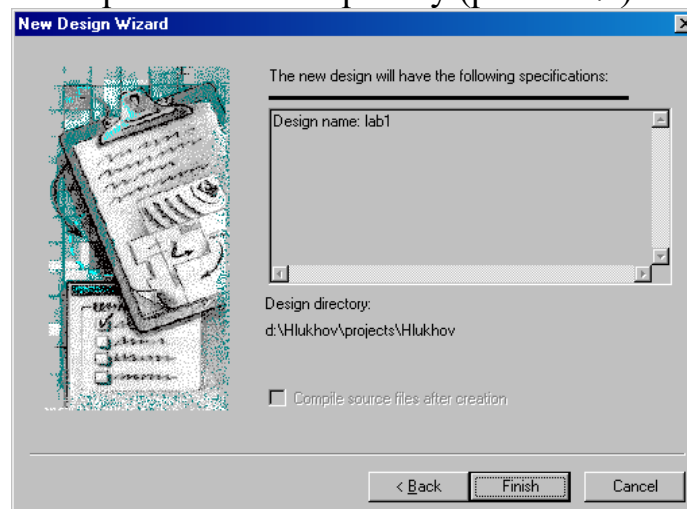


Рис. 1.17. Завершення створення нового проекту.