

## Лабораторна робота № 2

### ВИВЧЕННЯ СИСТЕМИ АВТОМАТИЗАЦІЇ ПРОЕКТУВАННЯ *QUARTUS II*. СТВОРЕННЯ ПРОЕКТУ

#### Ціль роботи:

- Вивчення системи автоматизації проектування *Quartus II*. Отримання навиків створення проекту, вводу проекту в схемотехнічному режимі, роботи в графічному редакторі, створення мегафункцій.
- Вивчення особливостей функціональної побудови суматорів. Розроблення функціональної схеми суматора в САПР *Quartus II*.

#### 1. Теоретичні відомості

Програмне середовище *Quartus® II* компанії *Altera®* є повною мультиплатформеною системою для автоматизації проектування (САПР), що містить набір інструментів для проектування цифрових пристроїв та систем на програмовному кристалі (*SOPC*). Середовище *Quartus II* включає в себе всі утиліти, необхідні для роботи з мікросхемами *FPGA* і *CPLD*. На рисунку 1.1 показані основні етапи проектування (*Design Flow*) в середовищі *Quartus II*.



Рис. 1.1. Етапи стандартного процесу проектування в САПР *Quartus II*

Стандартний процес проектування включає наступні етапи.

## I. Введення опису проекту

1. Створення нового проекту, вибір сімейства й типу мікросхеми. Використовується майстер створення проектів, який запускається командою **New Project Wizard** із меню **File**.

2. Введення опису проекту в САПР. Можна використовувати текстовий або графічний опис файлів проекту. За обома способами необхідно створити вихідні файли проекту, в тому числі й файл верхнього рівня ієрархії проекту (*Top Level*). За першим способом, вихідні файли створюються на мовах проектування апаратури *Verilog HDL*, *VHDL*, або *Altera Hardware Description Language (AHDL)* за допомогою текстового редактора (*Text Editor*). За використання графічного вводу блок-схема проекту створюється в графічному редакторі (*Block Editor*) з використанням уніфікованих символів, які представляють собою окремі блоки схеми, що описуються іншими вихідними файлами, та примітиви. В графічному режимі вводу проекту за допомогою майстра **MegaWizard® Plug-In Manager** (меню **Tools**) можна створювати різні мегафункції й IP-ядера і включати їх в файли проекту.

3. Вказання початкових налаштувань проекту. Використовується редактор призначень (**Assignment Editor**), діалогового вікна **Settings** (меню **Assignments**), редактор топології (**Floorplan Editor**), та/або технологія фіксованих логічних блоків (*LogicLock™*).

4. Проект може бути створено за допомогою спеціальних засобів для створення систем-на-кристалі (*SOPC Builder*) або для створення систем цифрової обробки сигналів (ЦОС) (*DSP Builder*). Програмні файли для процесорного ядра *Nios®* створюються за допомогою редактора програмного забезпечення (*Software Builder*).

## II. Синтез проекту

5. Синтез проекту. Використовується модуль аналізу й синтезу (**Analysis & Synthesis**).

6. Функціональне моделювання проекту. Використовується симулятор (**Simulator**) **и команда Generate Functional Simulation Netlist.**

7. Розміщення і трасування проекту. Використовується модуль трасування (**Fitter**).

8. Попередній аналіз споживаної потужності. Використовується утиліта **PowerPlay Power Analyzer**.

9. Аналіз часових затримок. Використовується утиліта аналізатор часових затримок (**Timing Analyzer**).

## III. Моделювання проекту

10. Моделювання проекту з урахуванням часових затримок. Використовується симулятор (**Simulator**).

11. Покращення часових характеристик проекту. Здійснюється повторний фізичний синтез проекту, налаштування в діалоговому вікні **Settings** та в редакторі призначень, фіксовані логічні блоки.

## IV. Програмування мікросхеми

12. Створення файлу для програмування пристрою на мікросхемі. Використовується модуль асемблера (**Assembler**).

13. Програмування мікросхеми за допомогою утиліти програматора (**Programmer**) й устаткування *Altera*; або перетворення формату файла для програмування.

## V. Внутрішньокристалне налагодження проекту

14. Налагодження проекту на мікросхемі. Використовується убудований логічний аналізатор (*SignalTap® II Logic Analyzer*), генератор контрольних крапок (*SignalProbe™*).

Проект є основина одиниця в САПР *Quartus II*. Проект створюється за допомогою майстра створення нового проекту **New Project Wizard**, що знаходиться в меню **File**. Майстер дозволяє задати робочу директорію, де будуть зберігатися файли проекту, призначити ім'я проекту і ім'я файлу верхнього рівня ієрархії проекту. До проекту можна підключити інші вихідні файли, користувацькі бібліотеки, САПР сторонніх компаній, що використовуються, обрати сімейство або тип мікросхем для реалізації проекту або використати автоматичний підбір мікросхем компілятором *Quartus II*.

### 2. Порядок виконання роботи

#### Частина 1. Створіть новий проект

1. Запустіть САПР *Quartus II*. Для виконання циклу лабораторних робіт використовується САПР *Quartus II Version 9.1 Build 222 10/21/2009 SJ Full Version*.

Запуск пакету виконується або з використанням іконки, розташованої на робочому столі комп'ютера, або з меню ПУСК: **Пуск** → **Всі програми** → **Altera** → **Quartus II Web Edition Full**. Для запуску з командного рядка наберіть **Quartus** і натисніть **Enter**. Після запуску з'явиться на екрані головне вікно пакету, яке має декілька робочих зон (рис. 1.2).

2. Створіть новий проект за допомогою майстра **New Project Wizard**.

Послідовність створення нового проекту:

2.1 Створіть в робочій директорії жорсткого диска папку з назвою проекту, наприклад *lab1\_SM*, інакше майстер запитатиме про створення нової папки на наступному кроці;

2.2. В меню **File** оберіть команду **New Project Wizard** (рис. 1.3). Відкриється вікно майстра. При першому запуску відображається вікно заставки (**Introduction**), в якому відображена послідовність дій, що супроводжує відкриття нового проекту. Якщо на екрані з'явиться сторінка **Introduction**, натисніть кнопку **Next**.

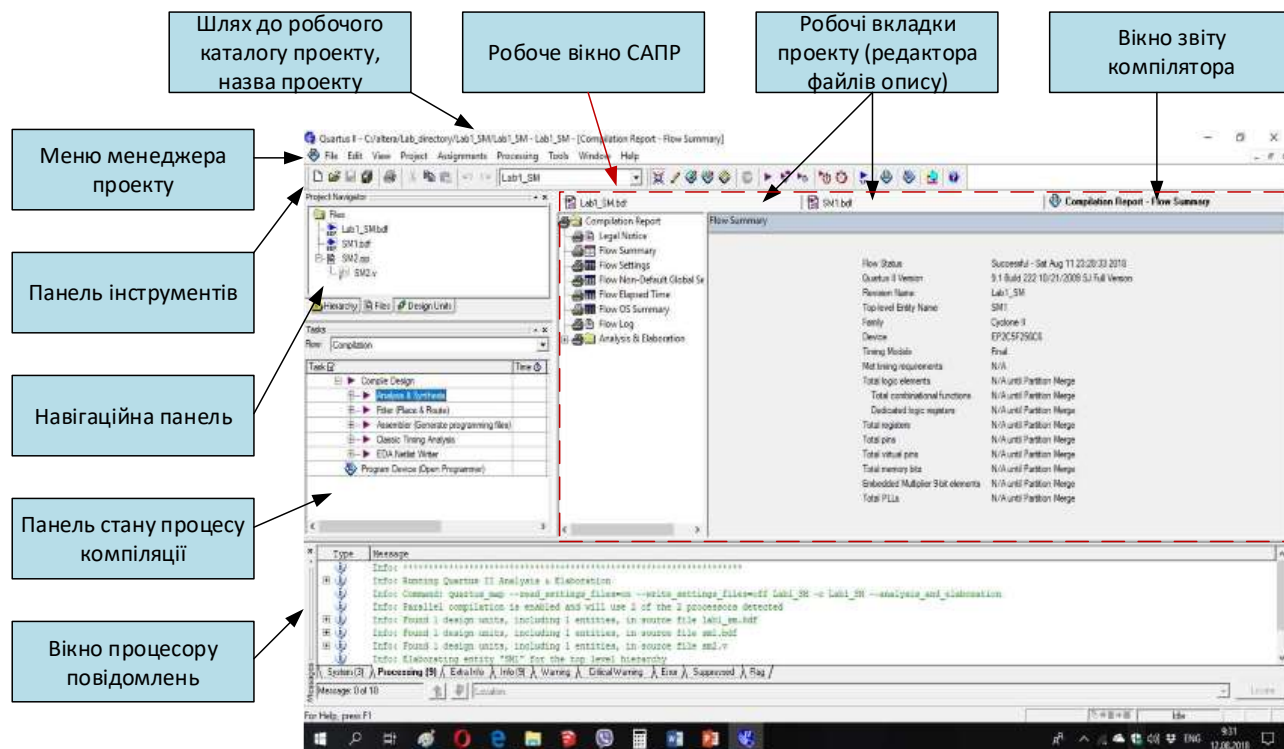


Рис. 1.2. Головне вікно середовища *Quartus II*.

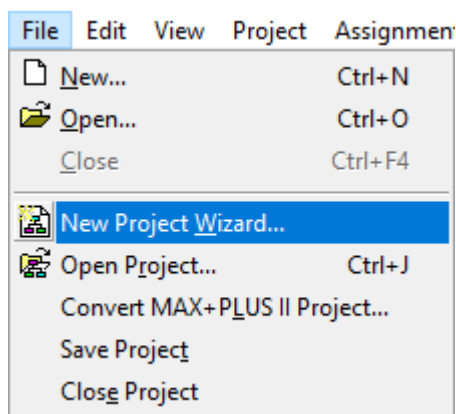


Рис. 1.3. Визивання майстра **New Project Wizard**


**2.3.** На першій сторінці майстра створення проекту (рис. 1.4) введіть наступну інформацію в текстові поля (приклад заповнення наведено в табл. 1):

Таблиця 1.1

Приклад для налаштування першої сторінки майстра **New Project Wizard**

Робочій каталог проекту	..\Lab_directory\Lab1_SM
Ім'я проекту	Lab1_SM
Ім'я об'єкту верхнього рівня розробки (файлу проекту)	Lab1_SM

- Назва робочого каталогу проекту. Каталог буде містити всі робочі файли проекту та інші файли, що пов'язані з даним проектом. Якщо каталог раніше не

був створений, *Quartus II* створить його автоматично. Якщо каталог вже був створений необхідно вказати шлях до нього, натиснувши на піктограму  праворуч від текстового поля (рис. 1.4). Введене ім'я автоматично відображається в інших текстових полях вікна;

- Ім'я проекту.

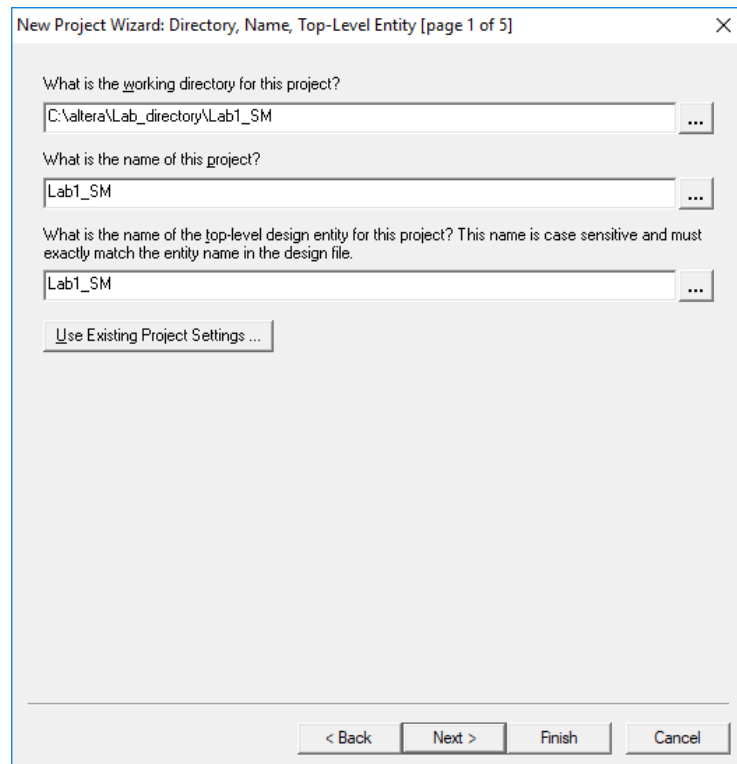



Рис. 1.4. Приклад налаштування першої сторінки майстра **New Project Wizard**

- Ім'я файлу верхнього рівня проекту. *Quartus II* автоматично створює установки компіляції та моделювання для файлу верхнього рівня проекту (головного файлу проекту) з зазначеним ім'ям. Задане ім'я проекту автоматично присвоюється головному файлу проекту в відповідному текстовому полі вікна. Припустимо задавати імена, відмінні від імені файлу проекту, але не рекомендується. Після створення проекту Ви зможете додати інші файли проекту і створити для них установки компіляції та моделювання за допомогою меню **Processing**.

Для переходу в наступне вікно натисніть кнопку **Next**;

**2.4.** Друге вікно майстра створення проекту (рис. 1.5) призначене для підключення до нового проекту раніше створених файлів. Для цього натисніть піктограму  і оберіть файл, який Ви хочете додати до Вашого проекту. Натисніть кнопку **Open**, після чого кнопку **Add**. Якщо Ви хочете підключити до проекту декілька файлів, виберіть їх імена і натисніть кнопку **Add All**, щоб додати їх в робочий каталог проекту.

Файл верхнього рівня не треба додавати до проекту, цей файл знаходиться в робочій директорії проекту і автоматично доданий до проекту майстром на попередньому кроці. Додавання файлів і каталогів на цьому кроці

виконується, якщо файли не знаходяться в робочій директорії проекту, були створені раніш, наприклад у складі інших проектів. Якщо Ви не маєте файлів проекту в інших каталогах або файлів, ім'я яких не співпадають з імені проекту, то додавати файли у цьому вікні не обов'язково.

Слід звернути увагу, що до проекту додаються лише шляхи до приєднаних файлів, фізично файли знаходяться в своїх вихідних директоріях.

Окрім файлів на другій сторінці можна додати бібліотеки спеціалізованих функцій, вказавши шлях до них. Це блоки, або функції, які були розроблені у складі інших проектів або сторонніми розробниками.

Під час виконання даної лабораторної роботи непотрібно підключати додаткові файли і бібліотеки, таким чином, для завершення роботи з другим вікном натисніть кнопку **Next**.

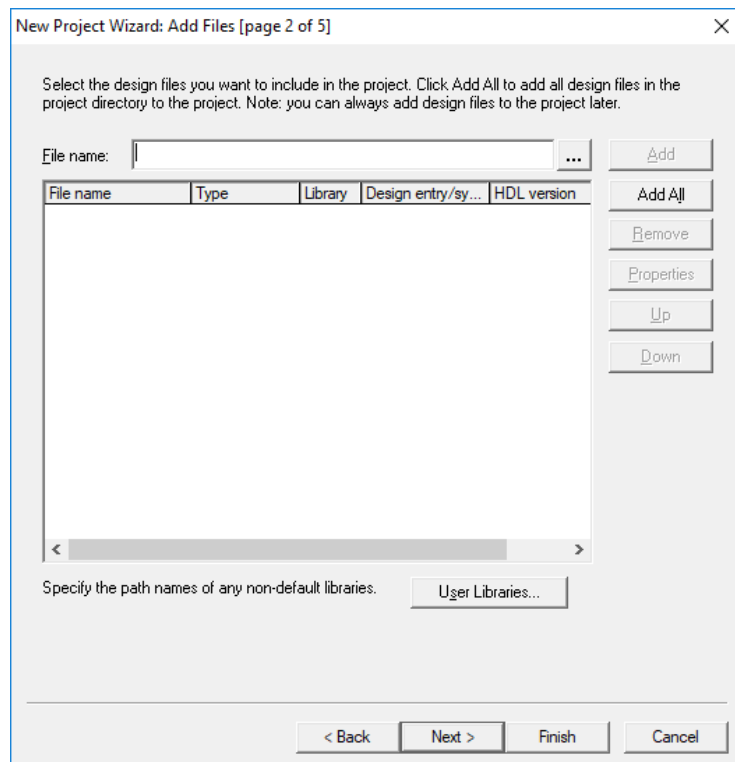


Рис.1.5. Налаштування другої сторінки майстра **New Project Wizard**

**2.5.** Третє вікно призначене для вибору сімейства і типу мікросхеми ПЛІС, які будуть використані під час компіляції проекту (рис.1.6).

Виберіть сімейство (**Family**) *Cyclone II*.

Для обмеження списку доступних мікросхем, можна використати спеціальні фільтри, якщо Вам відомі певні параметри мікросхеми. Наприклад, у частині вікна, що розміщена праворуч зверху, в розділі **Show in 'Available device' list**, встановіть наступні значення: в рядку **Package** оберіть **FBGA**, в рядку **Pin count** (кількість пінів) – **256**, **Speed grade** (швидкодія) – **Fastest**. У вікні **Available devices** оберіть мікросхему **EP2C5F256C6**. Натисніть **Next** для переходу на наступну сторінку.

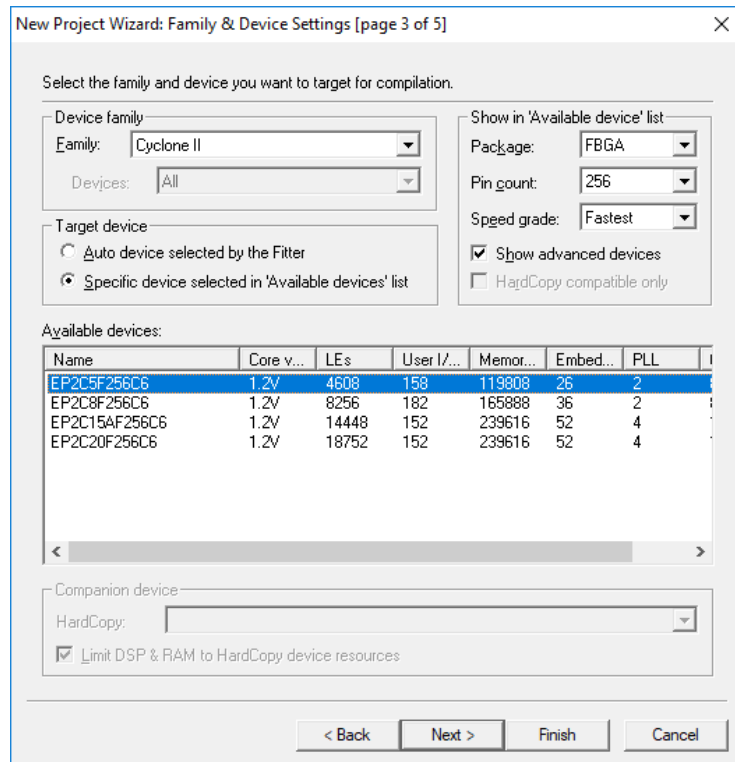


Рис. 1.6. Вибір типу мікросхеми

2.6. Четверте вікно майстра дозволяє указати САПР Quartus II додатково використовувані САПР сторонніх виробників під час створення даного проекту (рис. 1.7). Під час виконання лабораторних робіт використовується тільки САПР Quartus II, тому пропустить цей крок, натиснувши кнопку **Next** для продовження.

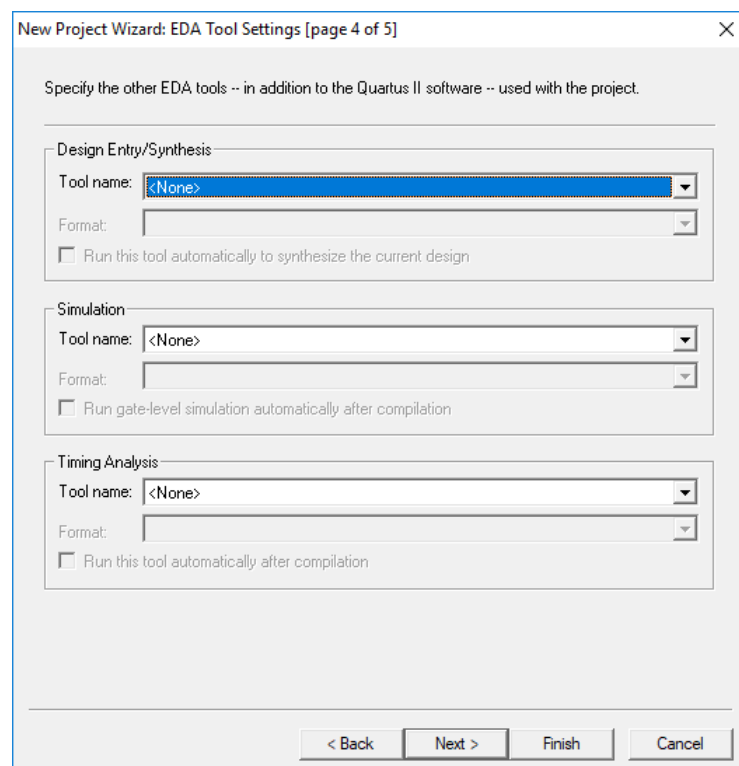


Рис. 1.7. Налаштування четвертої сторінки майстра **New Project Wizard**

2.7. П'ята сторінка містить повну інформацію про зроблені призначення. При



необхідності внесення виправлень, можна повернутися до попередніх сторінок, натиснувши кнопку **Back**.

На рис 1.8. представлена підсумкова сторінка. Натисніть кнопку **Finish**. Проект створений.

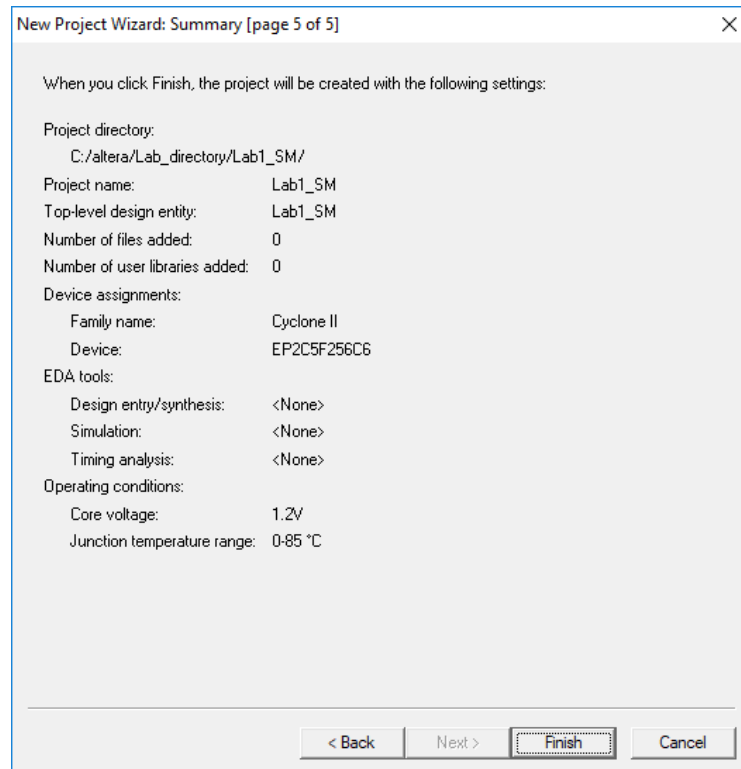


Рис. 1.8. Підсумкова сторінка роботи майстра створення проектів.

**2.8.** Для продовження виконання лабораторної роботи не потрібно виходити із середовища **Quartus II**. Після створення проекту в директорії проекту автоматично генерується ряд файлів, серед яких файл проекту (**Project File**) з розширенням **\*.qpf**. Для лабораторної роботи, що виконується, це файл **Lab1\_SM.qpf**. Закритий проект завжди можна відкрити за допомогою команди **Open Project** (меню **File**). У вікні провідника слід обрати каталог проекту і головний файл проекту (рис. 1.9).

Майстер **New Project Wizard** автоматично генерує наступні *файли з установками проекту*:

- Файл проекту (**Project File**) за замовчанням отримує ім'я **<ім'я проекту>.qpf**. Файл проекту зберігає конфігурацію проекту для компілятора. Під час проектування цей файл можна редагувати в текстовому редакторі.
- Файл з призначеннями проекту **<ім'я проекту>.qdf (Default File)**, — файл генерується і використовується за замовчанням якщо користувач не робить ніяких призначень під час створення проекту. Призначення, які зроблені користувачем зберігаються в файлі з призначеннями **<ім'я проекту>.qsf (Setting File)**, призначення можна редагувати прямо в файлі.

**2.9.** Після закінчення роботи з майстром **New Project Wizard** в основному вікні



програми **Project Navigator** в рядку **Hierarchies** з'явиться імена використовуваного сімейства ПЛІС і головний файл проекту. Одночасно у верхній частині основного вікна проекту з'являться шлях до проекту, назва проекту і файлу проекту (рис. 1.9\_)

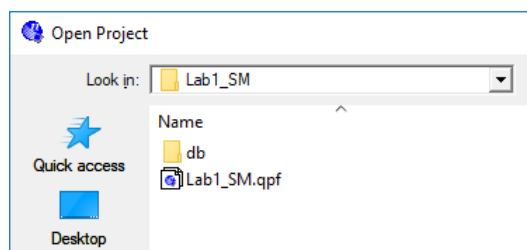


Рис. 1.9. Відкривання існуючого проекту

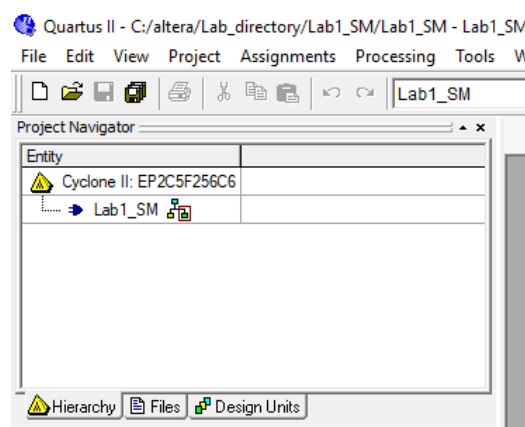
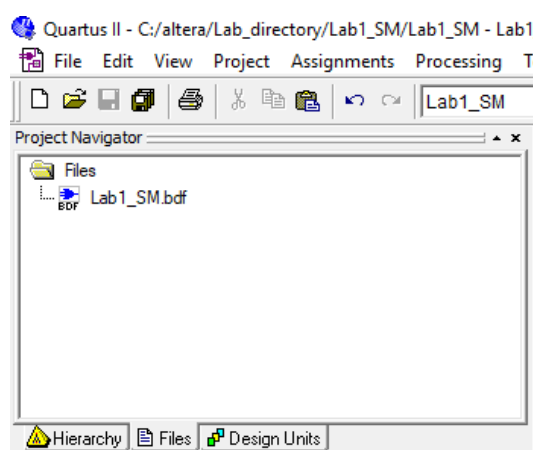


Рис. 1.9\_. Навігаційна панель після створення нового проекту



## Частина 2. Розробити модуль комбінаційного суматора

За варіантами завдань із табл. 1.2. синтезувати комбінаційну схему суматора. Теоретичні відомості для виконання синтезу суматорів наведені в [1, 2].

### *Приклад синтезу повного однорозрядного суматора*

Для прикладу виконаємо синтез повного однорозрядного суматора, що побудований на двох напівсуматорах. Функціональна схема такого суматора наведена на рис. 1.10.

### *Стислі теоретичні відомості для виконання завдання*

*Напівсуматором* називають комбінаційну схему, що реалізує функції  $q$  суми по  $\text{mod } k$  і переносу  $s$  при додаванні двох змінних.

Таблиця істинності напівсуматора наведена в табл. 1.3.

Таблиця 1.3

Таблиця істинності комбінаційного напівсуматора

$x$	$y$	$q$	$c$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Виходячи з таблиці істинності отримаємо наступні логічні вирази:

$$q = x \cdot \bar{y} \vee \bar{x} \cdot y = (x \vee y) \cdot (\bar{x} \vee \bar{y}) = (x \vee y) \cdot \bar{c}, \quad c = x \cdot y.$$

Повний однорозрядний суматор може бути побудований із двох напівсуматорів (рис. 1.10). Комбінаційна схема напівсуматора наведена на рис. 1.11.

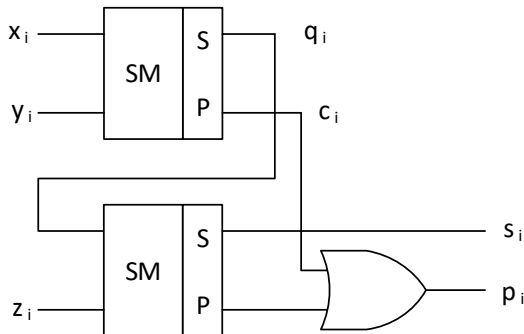


Рис. 1.10. Функціональна схема повного однорозрядного суматора, побудованого на двох напівсуматорах

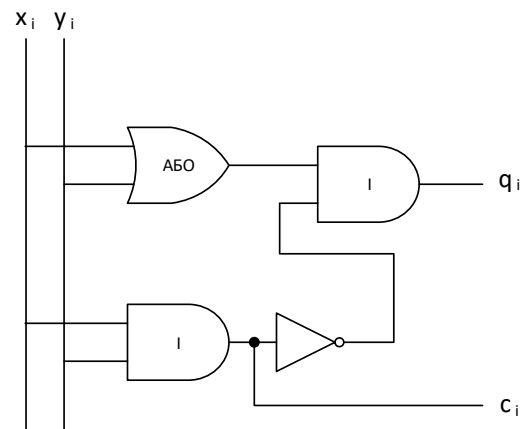


Рис. 1.11. Комбінаційна схема напівсуматора

**3.** Створіть файл верхнього рівня опису проекту. Для чого виконайте наступну послідовність дій:

*Файли опису проекту* – мають ієрархічну структуру і описують логіку проекту. Файли опису проекту можуть бути створені в графічному або текстовому редакторі користувачем. Файли створені в графічному редакторі мають певні користувацькі імена і розширення *\*.bdf*. Існує файл верхнього рівня ієрархії опису проекту, який не слід путати з головним файлом проекту. Файл проекту і файли з налаштуваннями не можуть бути файлами верхнього рівня опису ієрархії проекту.

Файл верхнього рівня опису проекту містить блок-схему верхнього рівня проекту – це блок, або декілька блоків, виводи яких приєднуються до виводів мікросхеми. Кожен блок може містити певну кількість вкладених блоків, які в свою чергу можуть мати ще вкладені блоки. Для опису вкладених блоків створюють файли опису проекту наступних рівнів ієрархії. Такий принцип опису проекту називається *структурним описом*.

**3.1.** В даному циклі лабораторних робіт для вводу проекту в САПР використовуємо графічний редактор. Якщо на навігаційній панелі (рис. 1.9) вибрати вкладку **Files**, ми побачимо, що у складі проекту немає жодного файлу опису проекту. Для створення файлу верхнього рівня ієрархії опису проекту

(оберіть в меню **File** команду **New**, відкриється вкладка **New**, де виберіть тип створюваного файлу (рис. 1.12). Необхідно вибрати вкладку **Design Files** (Файли опису проекту) і у списку, що випадає, обрати рядок **Block Diagram/Schematic File**. Після натискання кнопки **OK** відкриється вікно графічного редактора в головному вікні програми (рис. 1.13). В робочому полі графічного редактора вводять блок-схеми пристроїв та вузлів проекту. При створенні нового файлу проекту в графічному редакторі цей файл отримує ім'я за замовчанням (рис. 1.13). Збережіть створений файл опису проекту в каталозі проекту з ім'ям файлу проекту – **Lab1\_SM.bdf**. Під час першого збереження файлу опису САПР запропонує обрати саме це ім'я.

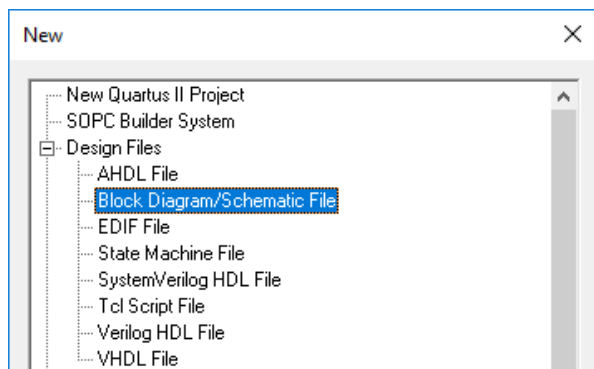


Рис. 1.12. Вікно вибору типу нового файлу проекту

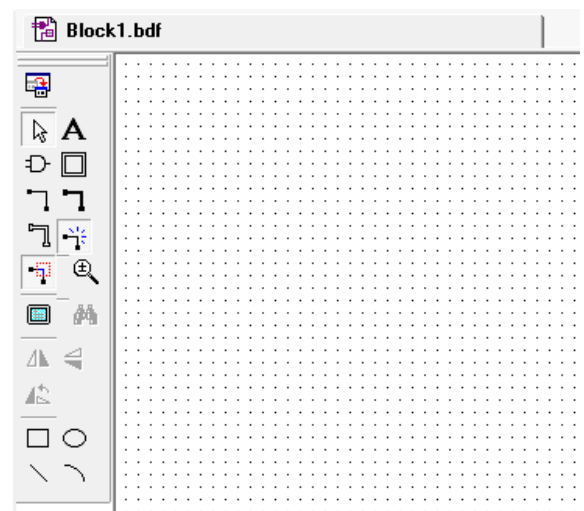


Рис. 1.13. Фрагмент вікна графічного редактора

Тепер на вкладці **Files** навігаційної панелі (рис. 1.9, а) можна побачити ім'я файлу верхнього рівня проекту. Можна закрити вікно графічного редактора і відкрити створений файл, натиснувши на імені файлу в навігаційній панелі.

У описаний вище спосіб створюють будь які файли опису проекту. Ієрархія файлів опису проекту відображується на вкладці **Files** навігаційної панелі вікна САПР.

Якщо непотрібно виконувати ніяких додаткових дій по обробці проекту, наприклад, компіляцію, можна використовувати команду **Open** із меню **File**, яка дозволяє відкрити окремий файл замість всього проекту цілком.

**3.2.** Введення блок схеми проекту виконується в робочому полі головного вікна з використанням панелі інструментів, призначення яких наведено на рис. 1.14.

**3.3** Виберіть піктограму «Створення блоку» (**Block Tool**) для створення робочого блоку. Натиснути на білій частині поля проекту ліву кнопку миші, і намалювати прямокутну область. Можна використовувати команди відміни (**Undo**) і повторення (**Redo**) дій із меню **Edit**. Вигляд створеного блоку показаний на рис. 1.15.


















Назва інструменту	Назва інструменту
 – від'єднання робочого вікна;	 – масштабування об'єктів;
 – виділення;	 – відображення робочого вікна на весь екран;
 – введення тексту;	 – пошук об'єкту;
 – введення стандартних компонент із бібліотеки	 – коригування відображення об'єкту;
 – створення блоку;	 – малювання прямокутника;
 – створення провідників;	 – малювання еліпса;
 – створення шин;	 – малювання прямих ліній;
 – створення каналів зв'язку;	 – малювання дуг.
 – автоматичне під'єднання провідників;	

Рис. 1.14. Призначення піктограм панелі інструментів графічного редактора

**3.4** Кожному блоку та елементу необхідно надати ім'я. Виберіть піктограму «Виділення» (**Selection Tool**) і клацніть правою кнопкою миші на створеному блоці. Відкриється діалогове вікно «Властивості блоку» (**Block Properties**), де на вкладці **General** в полі **Name** введіть ім'я блоку, наприклад, **SM1** (рис. 1.16). Поле **Instance name** можна залишити без змін, це внутрішній ідентифікатор елементу в проекті.

На вкладці **I/Os** призначте вхідні і вихідні сигнали блоку **SM1** (рис. 1.17) – в полі **Name** задайте ім'я сигналу, а в полі **Type** його тип: вхідний (**INPUT**), вихідний (**OUTPUT**) або двоспрямований (**BIDIR**). Для позначення проводника достатньо вказати лише ім'я сигналу, для позначення шини в квадратних дужках слід вказати її розрядність, наприклад **Data[3..0]**. Після заповнення текстових полів натисніть кнопку **Add** і перейдіть до внесення наступного сигналу. Після внесення всіх сигналів натисніть кнопку **OK**. Налаштований блок зображений на рис. 1.18.

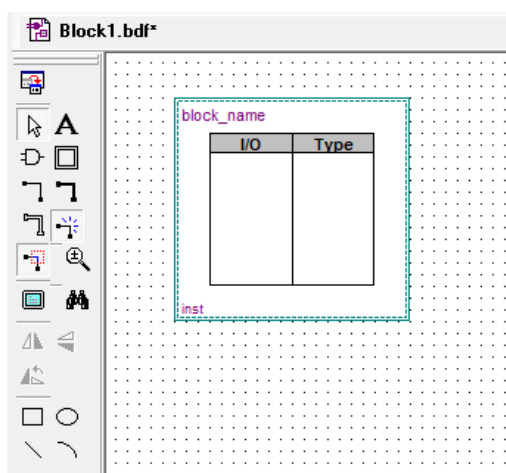


Рис. 1.15. Створення нового блоку

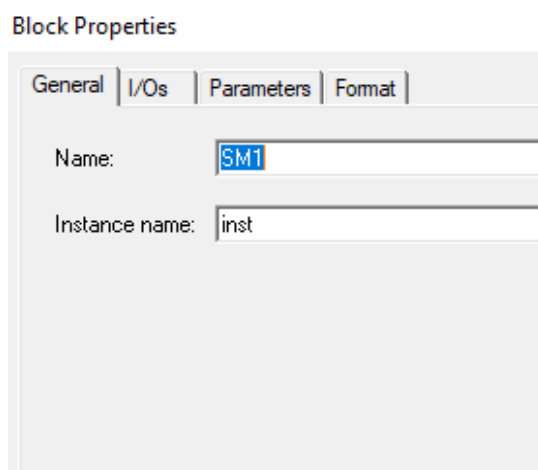


Рис. 1.16. Призначення ім'я новому блоку

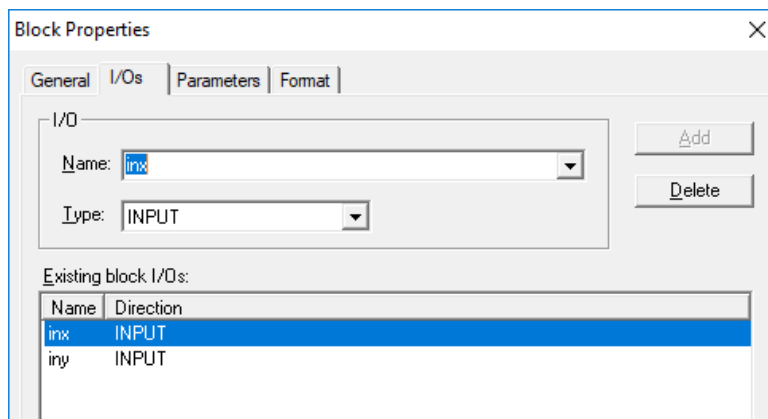


Рис. 1.17. Призначення сигналів введення/ виведення

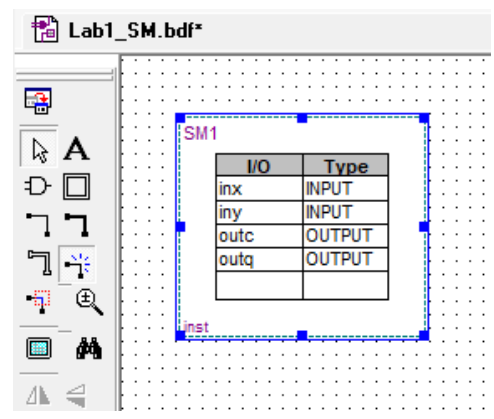


Рис. 1.18. Налагоджений блок

**3.5.** Додайте на робоче елементи функціональної схеми, яка зображена на рис. 1.10. В даному прикладі створимо блоки напівсуматорів двома способами, тому на цьому етапі замість другого напівсуматора схематично розмістить прямокутник, використавши інструмент «Прямокутник» (**Rectangle Tool**) панелі інструментів вікна **Symbol**.

Додайте на робоче поле графічного редактора логічний елемент АБО. Логічні елементи є примітивами *Quartus*.

*Примітив* Quartus це найдрібніша одиниця схеми, яка реалізована апаратно на кристалі FPGA. Будь яка схема, що введена користувачем в САПР, спеціальними засобами САПР буде перетворена у примітиви, які в подальшому будуть розміщені в комірки мікросхеми. Примітиви реалізовані в Quartus II, як бібліотечні компоненти, які знаходяться в бібліотеці стандартних компонентів. Для визивання бібліотеки стандартних компонентів натисніть піктограму «Інструмент для створення символу», після чого відкриється вікно «Символ» (**Symbol**), зовнішній вигляд, якого зображений на рис. 1.19.

Логічні елементи є бібліотечними компонентами, які знаходяться в бібліотеці стандартних компонентів Quartus II. Логічні елементи знаходяться в розділі *../libraries/primitives/logic*. Виберіть необхідний елемент із списку (для виконання завдання – логічний елемент АБО з двома входами **2or**). Зображення вибраного елемента відобразиться в головному полі вікна **Symbol** (рис. 1.20). Після натискання кнопки ОК вікно **Symbol** закривається, за натисканням у робочому полі редактора вибраний елемент відобразиться в визначеному місці. Встановлення прапорцю **Repeat-Insert Mode** дозволяє розмістити в робочому полі редактора декілька вибраних бібліотечних компонентів. Для відкріплення компонента від курсора миші натисніть правою кнопкою миші на робочому полі і виберіть команду **Cansel** в контекстному меню.

**3.6.** Створіть входні та вихідні контакти - піни (**Pins**) функціонального блоку. Контакти є бібліотечними елементами, які знаходяться в бібліотеці стандартних компонентів САПР. Контакти введення/виведення знаходяться в розділі *../libraries/primitives/pin* (рис. 1.19). Виберіть необхідний контакт із

списку контактів. Зображення вибраного елементу відобразиться в головному полі вікна **Symbol**. Натисніть кнопку OK і розмістити необхідну кількість контактів на робочому полі. Додайте три вхідних контакта (**Input**) для вводу аргументів і вхідного переносу і два вихідних (**Output**) для виводу результату підсумовування і вихідного переносу. Результат додавання контактів зображений на рис. 1.22.

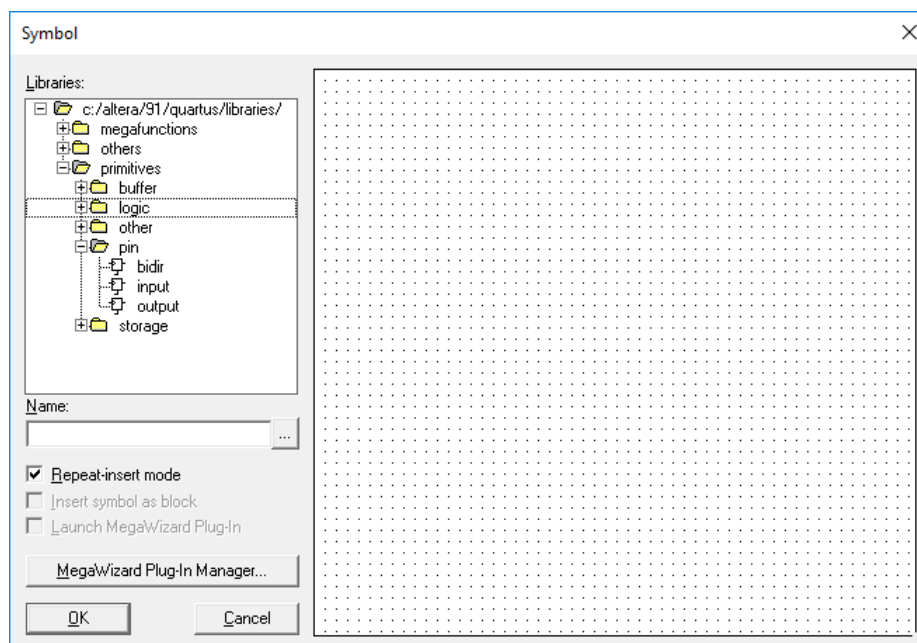


Рис. 1.19. Вікно інструменту для додавання бібліотечних символів функціональних елементів

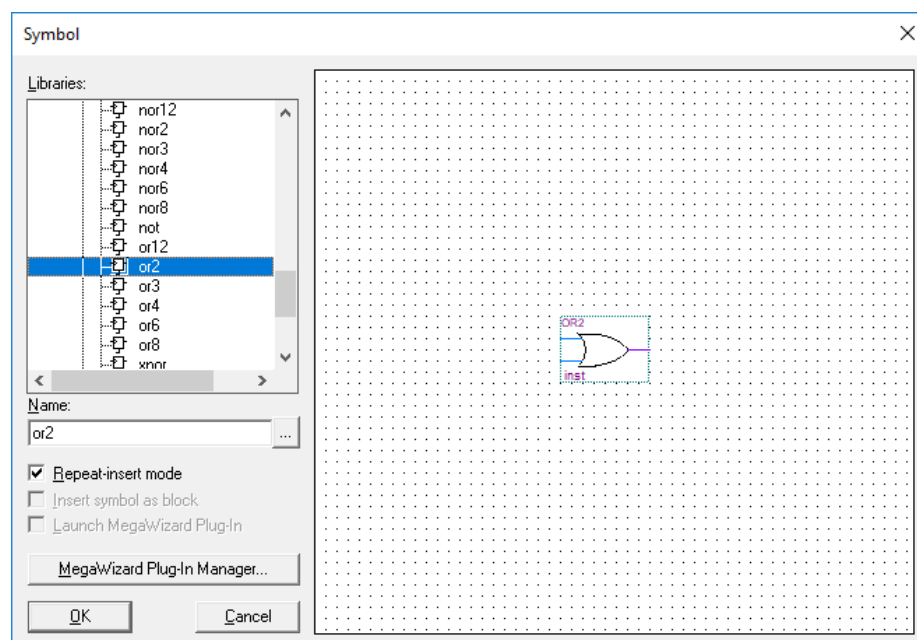


Рис. 1.20.

В результаті, на робочому полі створено фрагмент функціональної схеми суматора, який зображено на рис. 1.21.

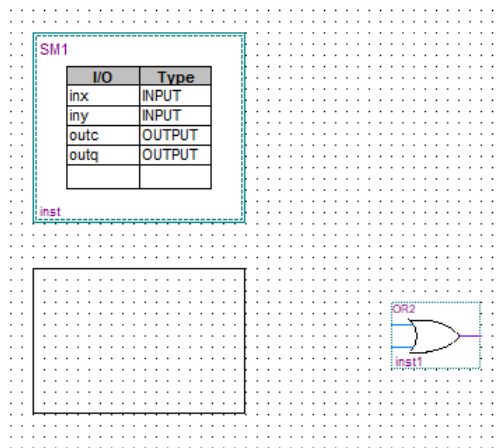


Рис. 1.21. Фрагмент функціональної схеми суматора

**3.7.** Призначте ім'я кожному контакту. Для цього натисніть правою кнопкою миші на одному з контактів. Оберіть команду **Properties** із контекстного меню. У вікні, що відкрилося, на вкладці **General** введіть ім'я контакту в текстовому полі **Name**. В полі **Default value** задається початкове значення сигналу: логічна одиниця **VCC** (встановлено за замовченням) або логічний нуль (**GND**). У випадку багаторозрядних контактів слід вводити ім'я контакту в форматі шини – із вказанням розрядності контакту в квадратних дужках, наприклад, **name[3..0]** або безпосередньо список всіх контактів. Аналогічно призначте ім'я всім контактам функціонального блоку згідно табл. 1.3 (див. рис. 1.22).

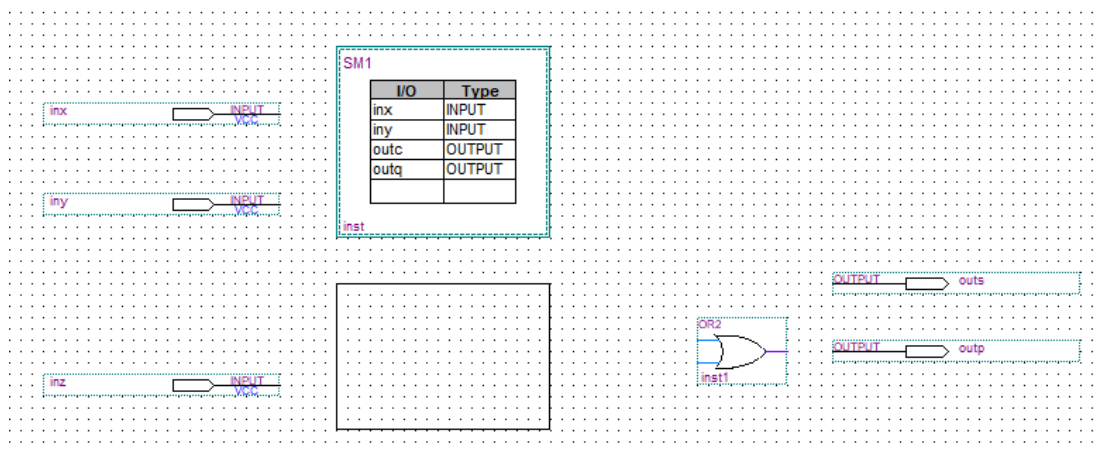


Рис. 1.22. Додавання контактів введення/виведення

Таблиця 1.3.

Приклад призначення імен контактам блоку суматора

Ім'я контакту (Name)	Тип контакту (Type)	Коментар
inx	INPUT	Доданок $x$
iny	INPUT	Доданок $y$
inz	INPUT	Вхідний перенос $z$
outs	OUTPUT	Сума $s$
outp	INPUT	Вихідний перенос $p$



**3.8.** Додайте зв'язки між функціональними вузлами суматора. Для створення провідників натисніть на піктограму визивання інструмента для створення провідників (**Node Tool**), для створення шин – піктограму визивання інструменту для створення шин (**Bus Tool**). Результат з'єднання зображено на рис. 1.23. Всім провідникам і шинам задайте ім'я. Для цього відкрийте вікно **Properties**, із контекстного меню кожної шини. Рекомендовано, щоб назва провідника/шини співпадала з назвою сигналу на виводі блоку.

САПР *Quartus II* припускає автоматичне з'єднання з використанням режиму *Smart* – якщо імена сигналів входів/виходів одного блоку збігаються з іменами відповідних сигналів іншого блоку, підключення виконується автоматично. Для створення з'єднань в явному вигляді використовуються карта з'єднань (*mappers*), при цьому імена виводів блоків ставляться у відповідність до імені сигналів на шинах і провідниках.

Для створення карти з'єднань натисніть правою кнопкою миші на одному з зелених прямокутників і місці приєднання провідника до блоку напівсуматора. Відкривається вікно «Властивості з'єднання» (**Mapper Properties**).

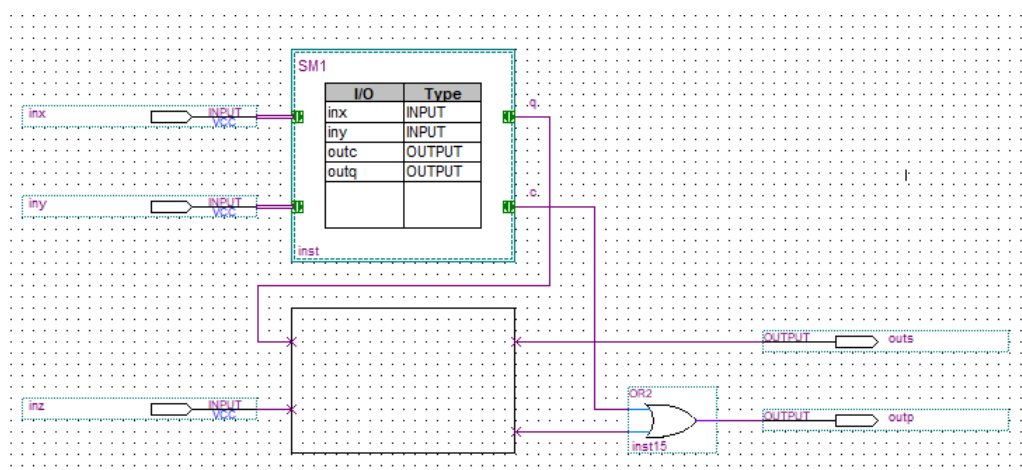


Рис. 1.23. Приклад створення з'єднань між функціональними елементами.

На вкладці **General** виберіть тип виводу блоку із списку **Type** (рис. 1.24). На вкладці **Mapping** задайте карту з'єднань. Для цього в випадаючому списку «Входи і виходи блоку» (**I/O on block**) виберіть ім'я виводу блоку. В випадаючому списку «Сигнали в каналі» (**Signals in conduit**) виберіть ім'я сигналу в лінії зв'язку. Натисніть кнопку **Add**, після чого відповідний запис з'явиться в полі «Наявні з'єднання» (**Existing mappings**) (рис. 1.25). Задайте всі інші карти з'єднань. Результат зображений на рис. 1.26.

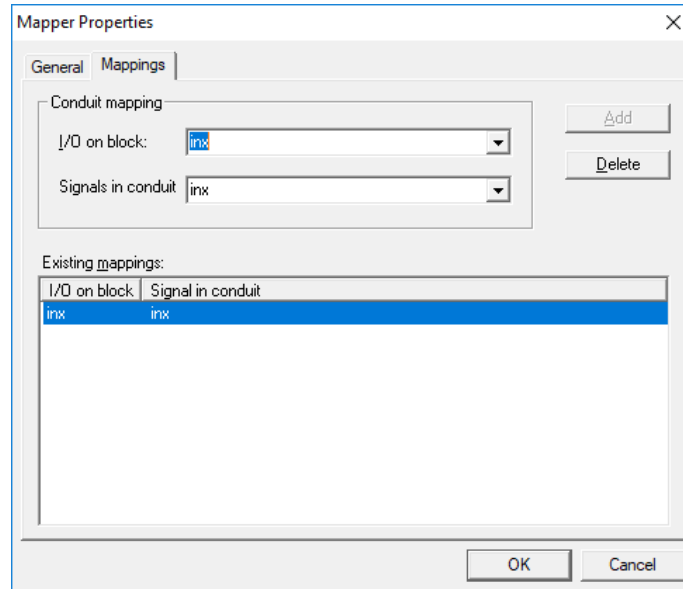


Рис.1.25.

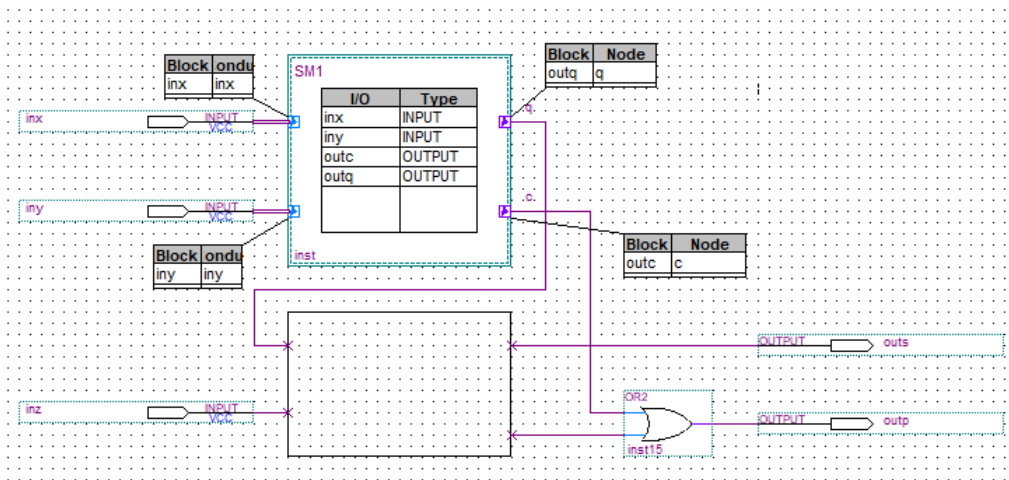


Рис. 1.26.

**3.9.** Створіть файли опису роботи блоку **SM1**. Для цього натисніть правою клавішею мишки на блоці напівсуматора та в контекстному меню виберіть команду «**Create Design File from Selected Block...**», відкриється вікно створення нового файлу опису виділеного блоку. В списку **File type** виберіть спосіб створення нового файлу, в нашому прикладі, це **Schematic** для створення файлу опису в графічному редакторі. Інші пункти списку призначенні для створення файлів опису на відповідних мовах програмування апаратури. Зверніть увагу, що САПР пропонує назву файлу опису відповідно до назви блоку, що описується. Можна задати власне ім'я файлу. Далі відкриється нова вкладка графічного редактора з заданим ім'ям (рис. 1.28). Новий файл з'явиться в навігаційній панелі (рис. 1.29).

Зверніть увагу, що знову створений файл опису містить контакти для приєднання до файлу верхнього рівня ієрархії, які були створені під час створення блоку верхнього рівня (рис. 1.28).

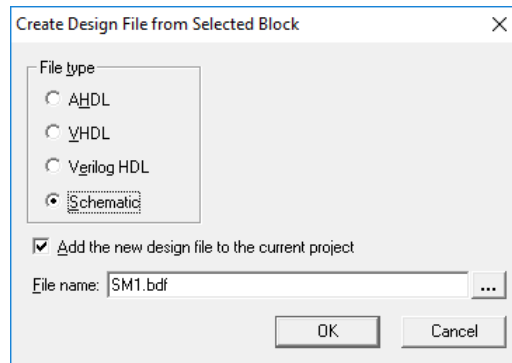


Рис.1.27.

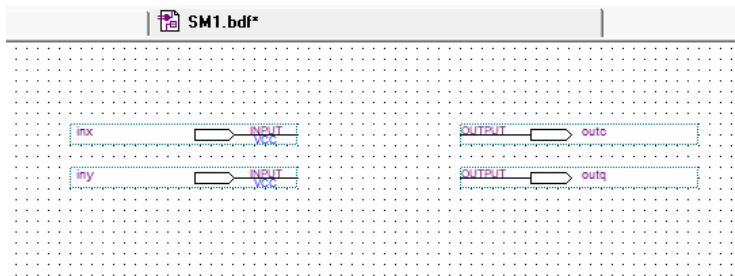


Рис. 1.28.

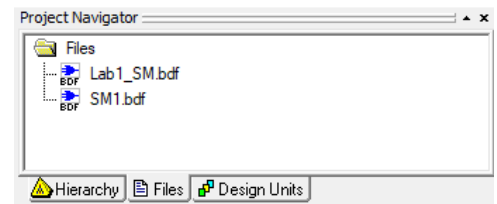


Рис. 1.29.

Розробіть на робочому полі функціональну схему напівсуматора, що зображена на рис. 1.11. Використайте стандартні компоненти: логічні елементи І, АБО. Збережіть створену схему. Результат зображений на рис. 1.29. Створений файл можна редагувати, відкривши файл із навігаційної панелі, або із контекстного меню, яке відкривається подвійним натисканням правої кнопки миші на блоці верхнього рівня опису проекту.

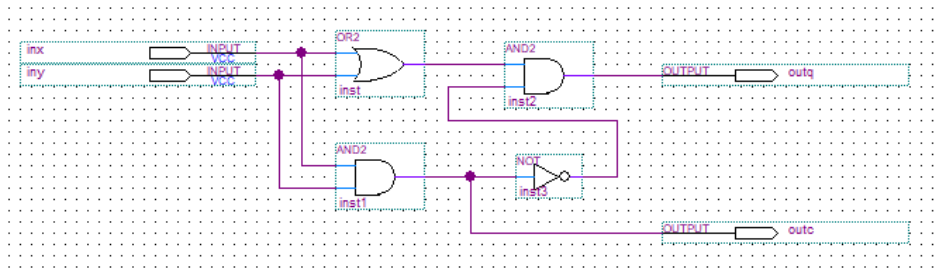


Рис. 1.29. Функціональна схема напівсуматора

Схематично ієрархічний принцип поєднання об'єктів проекту представлений на рис. 1.30.

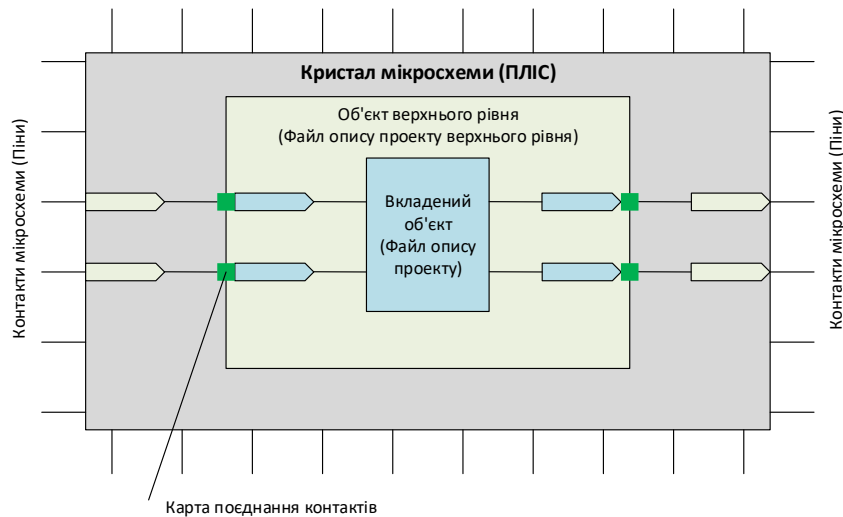


Рис. 1.30. Принцип ієрархічного поєднання об'єктів проекту

### 3.10. Створіть файли опису роботи блоку **SM2**.

Інший спосіб створення функціональних елементів це використання майстра **MegaWizard**. Цей майстер призначений для створення мегафункцій.

Для створення мегафункції напівсуматора виберіть команду **Mega Wizard plugin manager** в меню **Tools**. У вікні, що відкрилося (перша сторінка майстра), виберіть пункт **Create a new custom megafunction variation** (рис. 1.31). Натисніть кнопку **Next**, після чого відкриється вікно другої сторінки майстра **MegaWizard** (рис. 1.32).

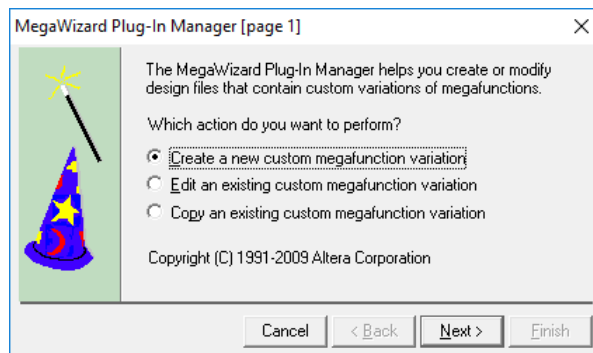


Рис. 1.31. Перша сторінка майстра **Mega Wizard plugin manager**

#### Базові бібліотеки САПР **Quartus II**

##### Мегафункції (Майстр **Mega Wizard plugin manager**)

- Мегафункції (**ALTPLL**, **ALTLVDS**, **ALTDDIO**...) – створені компанією Altera, дозволяють налаштування модуля відповідно певних вимог користувача і особливостей обраного сімейства мікросхеми.
- Бібліотека стандартних параметризованих модулів ( **LPMs** ):
  - Параметризовані логічні модулі (**LPM\_AND**, **LPM\_DECODE**...)
  - Параметризовані арифметичні модулі (**LPM\_ADD\_SUB**, **LPM\_COUNTER**...)

##### Бібліотека стандартних компонентів (Інструмент **Symbol Tool**)

- Бібліотека примітивів (**Primitives**: **AND**, **OR**, **INPUT**, **DFFE** ...);
- Бібліотека компонентів (серійні мікросхеми, налаштовані мегафункції, IP ядра);

Користувацькі бібліотеки (**User Libraries**).

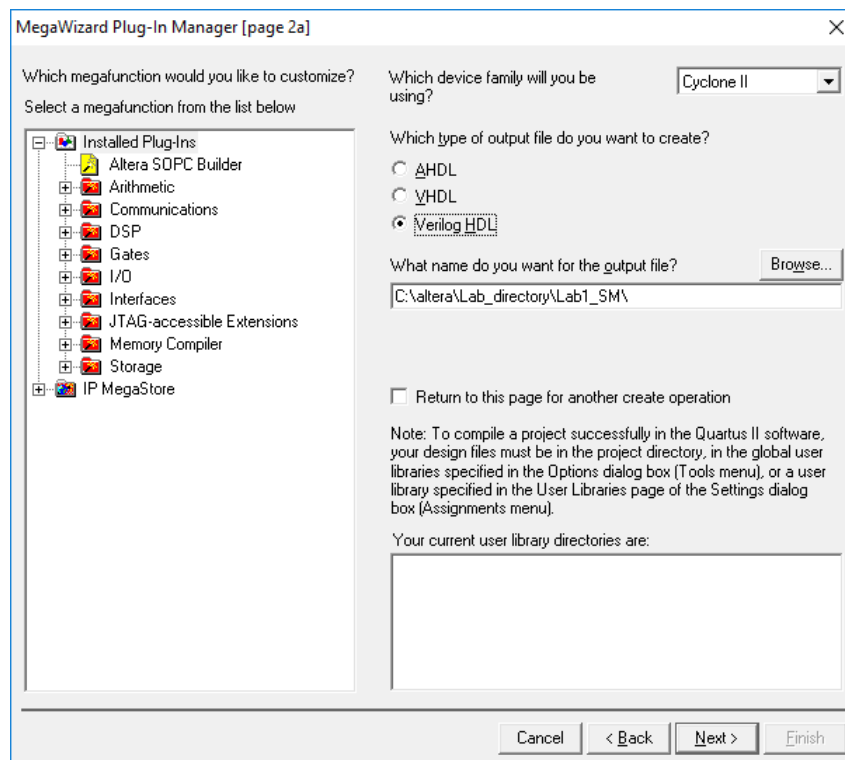


Рис. 1.32. Друга сторінка майстра **Mega Wizard**

Для створення мегафункції суматора налаштуйте наступні поля у другому вікні майстра:

- Встановіть прапорець для вибору мови опису інтерфейсу мегафункції, на якій буде додатково створено файл опису проекту – **Verilog HDL**.

Професійні розробники часто користуються графічним редактором і майстром **Mega Wizard** для створення файлів опису і налагодження і тестування функціональних блоків з подальшим створенням бібліотечних модулів. Майстер автоматично створює файл опису функціонального блоку на необхідній мові опису апаратури для використання у проектах.

- Вкажіть ім'я файлу опису блоку суматора (**SM2**) в полі **What name do you want for the output file?** Шлях до каталогу проекту вказаний у даному полі за замовчанням. Ви можете додати ім'я файлу в кінці обраного шляху до каталогу проекту або не вказувати повний шлях для автоматичного збереження файлу в робочій директорії проекту.

Якщо вказати ім'я без шляху, то файл буде створено в робочій директорії проекту і автоматично додано до проекту. Якщо ж вказати шлях - потім необхідно буде додавати файл опису до проекту вручну.

- Виберіть у списку бібліотечних компонентів пункт **Arithmetic**, де знаходяться мегафункції арифметичних операцій. Із бібліотеки параметризованих модулів (мегафункції з префіксом **LPM**) виберіть модуль суматора **LPM\_ADD\_SUB** (модуль може бути також налаштований як модуль для віднімання).

- Перевірте обране сімейство мікросхем, які будуть використані під час синтезу мегафункції, у текстовому полі у верхньому куті праворуч.

Указування сімейства мікросхем дозволяє утиліті **MegaWizard Plug-In Manager**

визначити доступні ресурси й доступні мегафункції для реалізації на даному кристалі. Можна створити мегафункцію для будь-якого сімейства мікросхем, для цього необхідно обрати відповідне сімейство із списку і створити для нього новий проект.

– Натисніть кнопку **Next**.

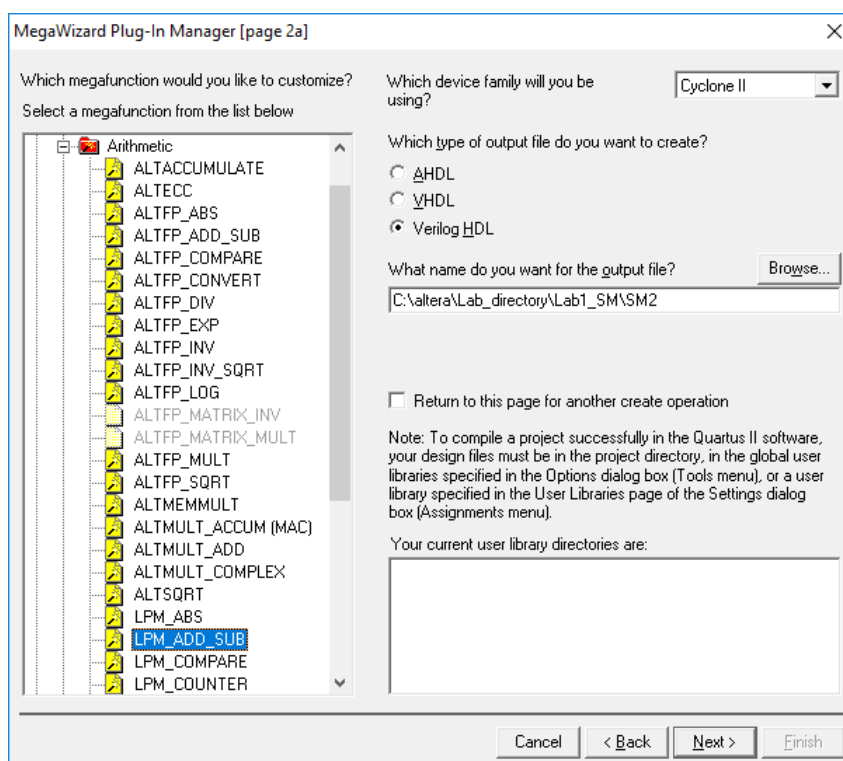


Рис. 1.33. Налаштування другої сторінки майстра **Mega Wizard**

У третьому вікні майстра (**General**) (рис. 1.34) налаштуйте розрядність шин напівсуматора і виконувану операцію:

Для створення мегафункції суматора налаштуйте наступні поля у другому вікні майстра:

- Оберіть розрядність аргументів, для виконання лабораторної роботи це один біт. Розрядність встановіть у списку, що випадає.
- Установленням прапорця, оберіть операцію, яку слід виконати, для виконання лабораторної роботи це операція додавання **Additional only**.
- На малюнку ліворуч зображено зовнішній вигляд блоку напівсуматора, який буде використано під час опису проекту.
- Натисніть кнопку **Next**.

У четвертому вікні майстра (**General 2**) (рис. 1.35) залиште налаштування за замовчанням, де вказано, що на вхідні шини суматора надходять змінні – **No, both values vary** (не константи) та буде виконуватись операції додавання без знаку – **Unsigned**. Натисніть кнопку **Next**.

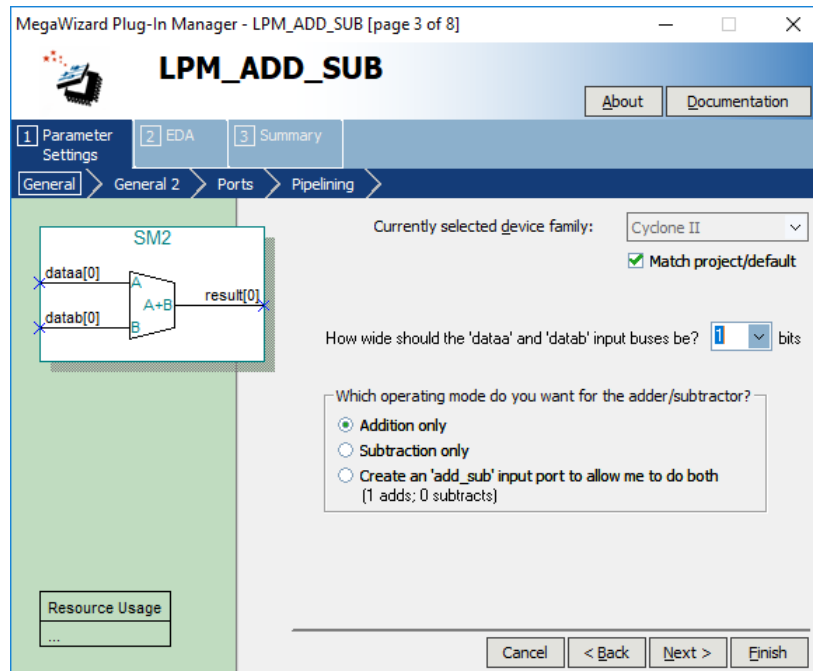


Рис. 1.34. Налаштування третьої сторінки майстра **Mega Wizard**

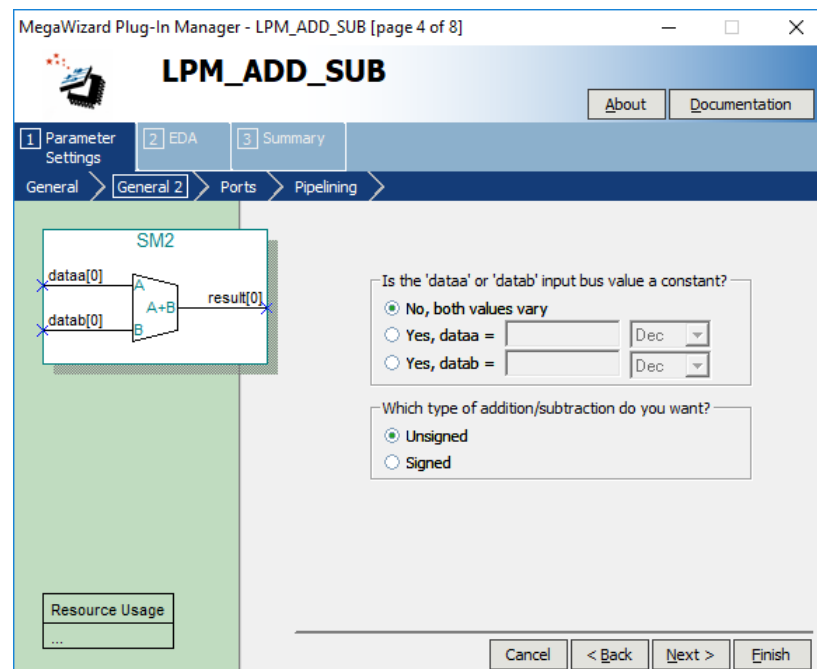


Рис. 1.35. Налаштування четвертої сторінки майстра **Mega Wizard**

У п'ятому вікні майстра (**Ports**) (рис. 1.36) додайте додатковий вихідний контакт для встановлення вихідного переносу – *Create a carry output*. Натисніть кнопку **Next**.

Для повного суматора можна додавати вхідний перенос та синхроімпульс для керування видаванням результату.



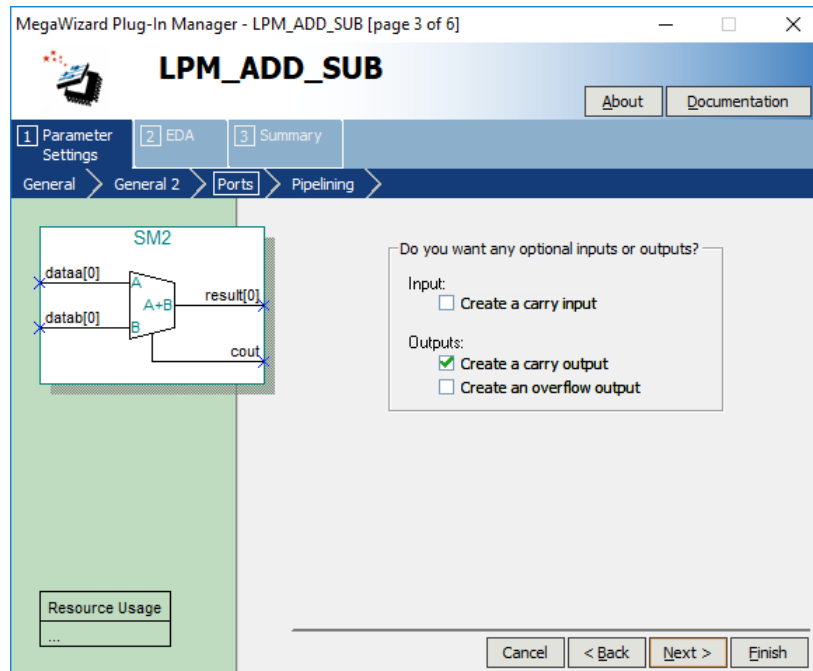


Рис. 1.36. Налаштування п'ятої сторінки майстра **Mega Wizard**

На шостій сторінці (**Pipelining**) виберіть пункт *Yes, I want an output latency of 2 clock cycles* (затримка видавання результату на 2 циклу тактової частоти) (рис. 1.37). Натисніть кнопку **Next**.

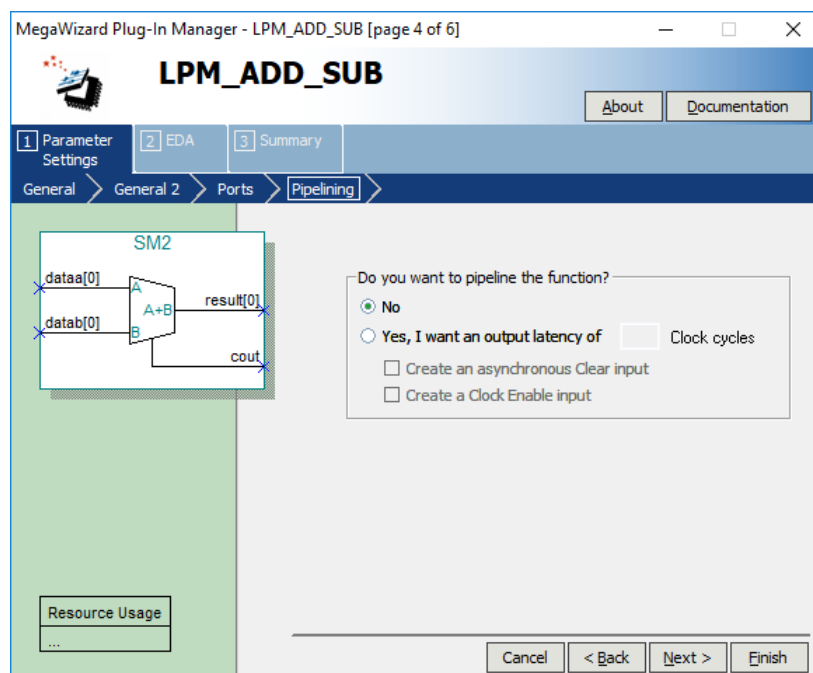


Рис. 1.37. Налаштування шостої сторінки майстра **Mega Wizard**

Надалі, на сьомій сторінці (друга вкладка **EDA**) вказують файл **Lpm**, який використовується сторонніми САПР для моделювання мегафункції, але в даній лабораторній роботі сторонні САПР не використовуються, тому пропустить цю сторінку. Натисніть кнопку **Next**.

На восьмій сторінці (третя вкладка **Summary**) відмітьте файли, що повинні бути створені майстром **Mega Wizard**. Для файлів типу *Scematic* це файли **SM2.v** (якщо для опису інтерфейсу мегафункції обрано мову **Verilog**) і

**SM2.bsf** (файл із графічним зображенням (символом) блоку мегафункції) (рис. 1.38).

Натисніть кнопку **Finish** для закінчення створення мегафункції напівсуматора.

Якщо мегафункція створена некоректно або Ви хочете внести будь-які зміни відкрийте заново майстер **MegaWizard Plug-In Manager** із меню **Tools**. Оберіть опцію редагування мегафункції **Edit an existing megafunction variation** і винесіть корективи. Натисніть кнопку **Finish**.

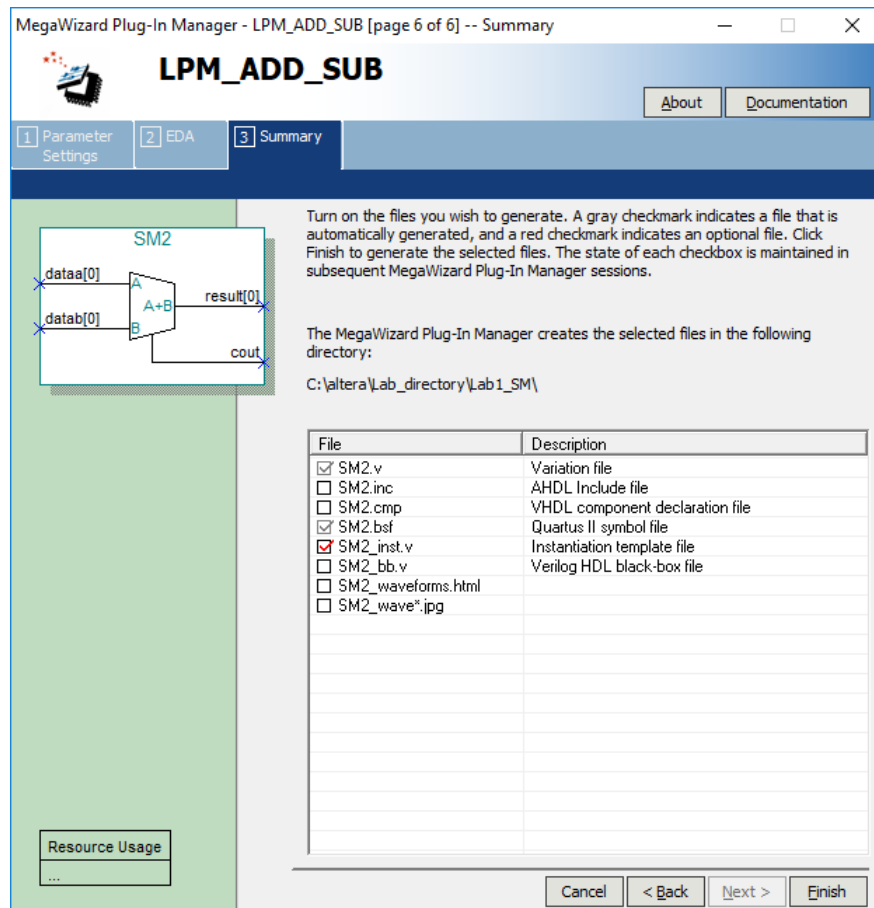



Рис. 1.38. Налаштування останньої сторінки майстра **Mega Wizard**

### 3.11. Додайте блок напівсуматора в проект і створіть необхідні зв'язки.

Відкрийте файл **Lab1\_SM.bdf**, для цього можна використати команду **Open** в меню **File**, натиснути піктограму  на панелі інструментів або два рази натиснути на імені файлу у вікні навігаційної панелі **Project Navigator**. Якщо файл не був закритий, його можна знайти на одній із вкладок робочого поля вікна САПР.

Відкрийте вікно **Symbol** за допомогою відповідної піктограми, або за допомогою подвійного натискання мишею в будь-якому вільному місці робочого поля. У вікні оберіть вкладку **Project**. В цій папці розміщуються всі користувацькі мегафункції, створені в рамках проекту. Двічі натисніть лівою кнопкою миші на елементі **SM2**. Натисніть ліву кнопку миші в полі креслення і розмістить вибраний елемент замість прямокутника на функціональній схемі.

