

# Лабораторна робота 1

## Графічне проектування арифметико-логічного пристрою

### 1 Мета лабораторної роботи:

оволодіти знаннями і навичками по проектуванню арифметико-логічних пристроїв (АЛП) для сучасних комп'ютерів. Знайомство з САПР Aldec ActiveHDL, навчитись вводити логічні схеми АЛП в графічному редакторі.

### Теоретичні відомості

АЛП призначений для виконання як арифметичних дій (додавання, віднімання) так і логічних дій (побітове І, АБО, НІ, Виключне АБО) над даними, які представлені паралельними  $n$ -розрядними кодами з фіксованою комою, наприклад,  $A$  і  $B$ . Частіше всього ці дані представлені в доповнюючому коді. Як особливий операнд виступає біт  $C_0$  переносу в молодший розряд. Крім  $n$ -розрядного результату  $Y$ , результатами АЛП часто є такі ознаки результату, як перенос з старшого розряду  $C_n$ , переповнення  $V$ , ознака нульового результату  $Z$  і біт знака  $S$ .

Тип операції АЛП задається керуючим кодом  $F$ , кодування якого вибирається в кожному випадку особливо.

### 2. Завдання для лабораторної роботи:

- розробити функціональну схему АЛП, що виконує задані функції;
- змодельовати роботу АЛП.

**Результати виконання** оформлюються у вигляді звіту (протоколу). Звіт повинен вміщувати:

- опис і рисунок заданого варіанта АЛП,
- хід проектування і схему АЛП,
- графіки сигналів, знятих при іспитах АЛП,
- висновки.

У всіх варіантах завдань необхідно обчислити деяку функцію  $Y=F(A,B)$ , біт  $Z$  – ознаку нульового результату і біт  $C_2$  – перенос з старшого розряду.

Варіант завдання вибирається за номером студента в списку групи з наступної таблиці

№ завдання	Функції $Y$ в залежності від $F$	№ завдання	Функції $Y$ в залежності від $F$
1	$A+B+C_0$ , $A$ and $B$	12	$A-B-C_0$ , $A$ nand $B$
2	$A+B+C_0$ , $A$ or $B$	13	$A-B-C_0$ , $A$ nor $B$
3	$A+B+C_0$ , $A$ xor $B$	14	$A-B-C_0$ , $A$ xor $B$
4	$A+B+C_0$ , $A$ xnor $B$	15	$A-B-C_0$ , $A$ xnor $B$
5	$A+B+C_0$ , $\max(A,B)$	16	$A-B-C_0$ , $\max(A,B)$
6	$A+B+C_0$ , $\min(A,B)$	17	$A-B-C_0$ , $\min(A,B)$
7	$A+B+C_0$ , $A-B-C_0$	18	$A-B-C_0$ , $\text{abs}(A)$
8	$A+B+C_0$ , кількість нулів перед першою 1 в $A$ (пріоритетний шифратор)	19	$A-B-C_0$ , кількість нулів перед першою 1 в $A$ (пріоритетний шифратор)
9	$A+B+C_0$ , кількість одиниць разом в $A$ і $B$	20	$A-B-C_0$ , кількість одиниць разом в $A$ і $B$
10	$A+B+C_0$ , кількість нулів разом в $A$ і $B$	21	$A-B-C_0$ , кількість нулів разом в $A$ і $B$
11	$A+B+C_0$ , $\text{abs}(A)$	22	$A-B-C_0$ , $\text{abs}(B)$

### 3. Виконання роботи

Спочатку складають таблиці істинності вихідних сигналів АЛП. Для кожного розряду АЛП складають окрему таблицю. Це необхідно для того, щоб спробувати виявити закономірність в побудові логічної схеми одного розряду в багаторозрядному АЛП.

За таблицями істинності записують Булевські рівняння для вихідних сигналів.

Булевські рівняння використовують для накреслення функціональної схеми. При цьому використовують бібліотеку символів логічних елементів, вбудовану в САПР ActiveHDL. В табл. 1 представлена частина цієї бібліотеки




Табл.1. Бібліотека символів логічних елементів



Символ	Функція	Символ	Функція
AND2	2-входова &	OR2	2-входова ∨
AND2B1	2-входова & з 1 інверсним входом	OR2B1	2-входова ∨ з 1 інверсним входом
AND2B2	2-входова & з 2 інверсними входами	OR2B2	2-входова ∨ з 2 інверс. входами
AND3	3-входова &	OR3	3-входова ∨
AND3B1	3-входова & з 1 інверсним входом	OR3B1	3-входова ∨ з 1 інверсним входом
AND3B2	3-входова & з 2 інверсними входами	OR3B2	3-входова ∨ з 2 інверс. входами
AND4	4-входова &	OR4	4-входова ∨
NAND2	2-входова &-not	XOR2	2-входова ⊕
NAND2B1	2-входова &-not з 1 інверс. входом	XOR3	3-входова ⊕
NAND2B2	2-входова &-not з 2 інверс. входами	XNOR2	2-входова ⊕-not
NAND3	3-входова &-not	XNOR3	3-входова ⊕-not
NAND3B1	3-входова &-not з 1 інверсним входом	BUF	Буфер - повторювач
NAND3B2	3-входова &-not з 2 інверс. входами	BUFT	Тристабільний буфер
NAND4	4-входова &-not	INV	Буфер - інвертор

Запускають САПР ActiveHDL, відкривають власний робочий простір Workspace, вводять назву проекту, наприклад, ALU\_lab1.



Натиснувши праву кнопку миші на строці Add\_New\_File вибирають опцію New -> Block\_Diagram.





В запропонованому діалозі вводять назву проекту і назви портів Entity.



У вікні Diagram за допомогою засобів графічного вводу ввести функціональну схему, що відповідає одержаним Булевським рівнянням. При цьому бібліотеку символів відкривають натиснувши піктограму , компоненти з бібліотеки вводять перетаскуванням їх зображення з вікна Symbols Toolbox в поле креслення. Багаторозрядні шини креслять, вибравши піктограму жгута , а лінії зв'язку – піктограму лінії . Вибравши елемент креслення вказівкою курсором і натиснувши праву кнопку миші, вибирають меню Properties, в якому задають необхідні властивості елемента. Так, вибравши лінію, задають її ім'я – Net Name в полі Segment, яке є ім'ям відповідного сигналу. Ім'я лінії з вказівкою її номера, яка входить в багаторозрядну шину задавати обов'язково, наприклад, як A(1).

Після закінчення вводу схеми її транлюють в VHDL- програму, натиснувши піктограму . Якщо в схемі є помилки, про що буде повідомлення у вікні консолі, ці помилки необхідно усунути. Після успішної компіляції одержану VHDL-програму можна побачити, натиснувши .

Остаточну компіляцію VHDL-програми виконують натиснувши .

Одержану модель АЛП моделюють після виклику вікна графіків сигналів . В цьому вікні встановлюють імена сигналів, які необхідно відобразити, в тому числі всі вхідні сигнали викликавши вікно . Можна встановити сигнали, перетаскуючи їх з вікна Design Browser – Structure.

Всі вхідні сигнали необхідно підключити до стимуляторів. Для цього викликають вікно стимуляторів . Серед стимуляторів можна вибрати генератор синхросигналів з заданим періодом , лічильник , натискання вибраної кнопки клавіатури , або задану послідовність зміни (графік) сигналу (Formula).

Моделюють проект, задавши інтервал часу моделювання  і натиснувши кнопку . Симулятор встановлюють в початковий стан, натиснувши .

Одержані графіки сигналів заносять в протокол лабораторної роботи.

#### 4 Приклад виконання роботи

Розглянемо приклад проектування  $n = 2$  –розрядного АЛП, що виконує функції

$Y = A - B$  при  $F=0$ ,  $Y = ($ кількість одиниць в числі  $A)$  при  $F=1$ .

При цьому крім результату  $Y$ , видається біт  $Z$  – ознака нульового результату і біт  $C_2$  – перенос з старшого розряду.

Логіка АЛП виражається наступними таблицями істинності

F	A0	B0	C0	A1	Y0	C1
0	0	0	0	x	1	0
	0	0	1	x	0	1
	0	1	0	x	0	0
	0	1	1	x	1	0
	1	0	0	x	0	1
	1	0	1	x	0	1
	1	1	0	x	1	0
	1	1	1	x	0	1
1	0	x	x	0	0	0
	0	x	x	1	1	0
	1	x	x	0	1	0
	1	x	x	1	0	1

F	A1	B1	C1	Y1	C2
0	0	0	0	1	0
	0	0	1	0	1
	0	1	0	0	0
	0	1	1	1	0
	1	0	0	0	1
	1	0	1	0	1
	1	1	0	1	0
	1	1	1	0	1
1	X	X	0	0	0
	X	X	0	0	0
	X	X	1	1	0
	X	X	1	1	0

Перша і друга таблиці виражають логіку першого і другого розрядів АЛП, відповідно. Тут  $x$  означає байдужий стан.

Функціонування АЛП описується наступними Булевськими рівняннями, що виходять з таблиць істинності.

$$Y0 = (\bar{F}(A0 \oplus B0 \oplus C0)) \vee (F \& (A0 \oplus A1));$$

$$C1 = (\bar{F}(B0C0 \vee A0C0 \vee A0B0) \vee F(A0A1));$$

$$Y1 = (\bar{F}(A1 \oplus B1 \oplus C1)) \vee (FC1);$$

$$C2 = \bar{F}(B1C1 \vee A1C1 \vee A1B1);$$

$$Z = \bar{Y}1\bar{Y}2.$$

Побудовані рівняння реалізуються функціональною електричною схемою, яка викреслюється за допомогою графічного редактора ActiveHDL. Одержана схема показана на рис.1.

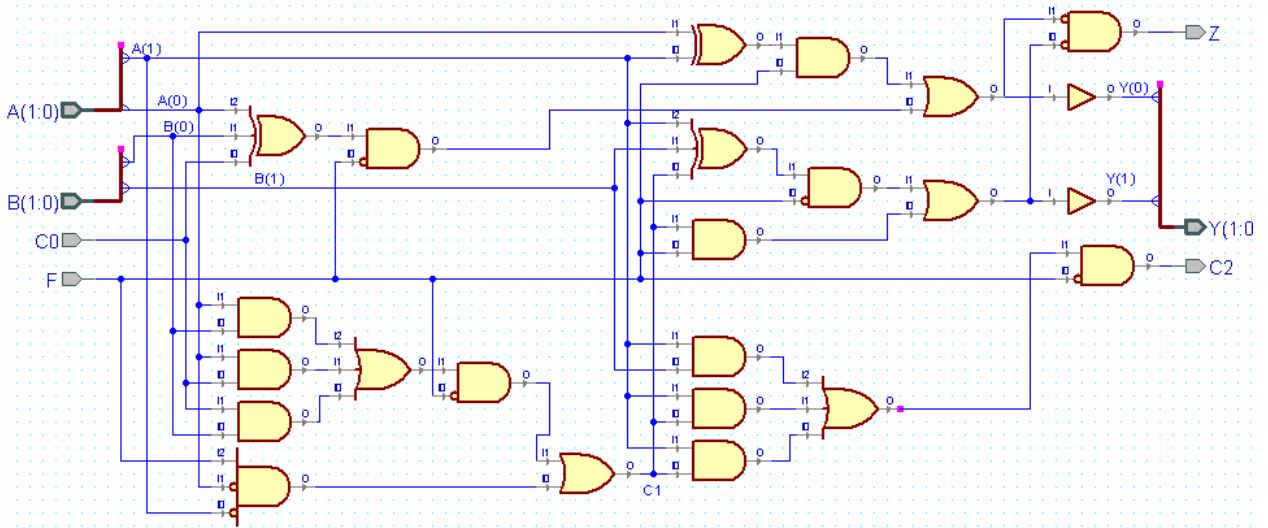


Рис.1. Функціональна схема АЛП

Результати моделювання АЛП у вигляді графіків сигналів показані на рис.2.

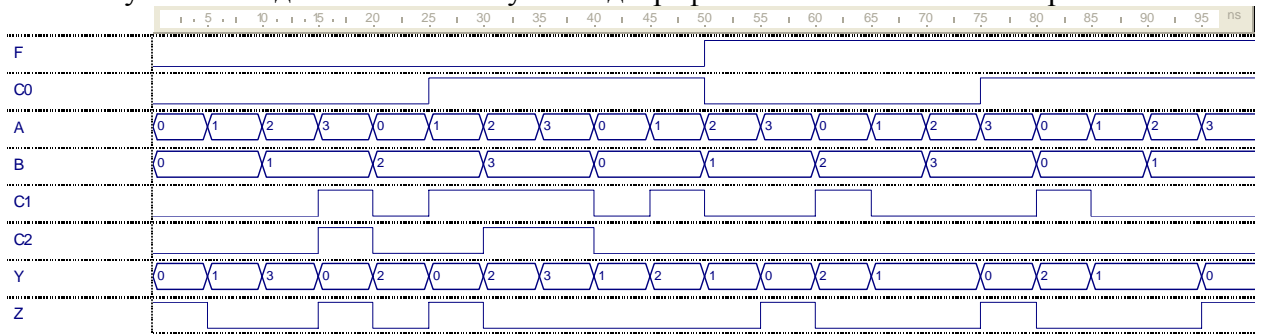


Рис.2. Графіки вхідних і вихідних сигналів АЛП