ЛЕКЦІЯ 4
ОСНОВНІ ПОНЯТТЯ І ВИЗНАЧЕННЯ АЛГЕБРИ ЛОГІКИ

1.1 Основні визначення алгебри логіки

Логіка - це наука про закони і форми мислення. Математична логіка - наука про застосування математичних методів логіки при аналізі і синтезі логічних завдань у питаннях вивчення основ математики.

Ідея побудови універсальної мови математики висувалася ще в 17 столітті Лейбніцем. Але тільки в 19 столітті Дж. Буль (1847р.), Пірс (1885р.), де Морган (1958р.) увели в мову математичної логіки предикати (функції F, які приймають значення «0» або «1») предметні змінні (звичайно x_1, x_2,…x_n, від яких залежить F) і квантори (описують характер залежності F від x_1, x_2,…x_n). Після чого була почата спроба зведення всієї математики до логіки (Пост, Шеннон, Шестаков, Глушков, Яблонский, ін.). Спроба, однак, не увінчалася повним успіхом, тому що виявилося не завжди можливим вивести із чисто логічних аксіом існування деяких понять на нескінченній множину [1].

Ми будемо розглядати найбільш елементарний розділ математичної логіки, що зветься ***алгебра логіки.*** Інші найбільше часто вживані назви: **булева алгебра, алгебра висловлень, бінарна логіка.**

Алгебра логіки - розділ математичної логіки, що вивчає висловлення, розглянуті з боку їхніх **логічних значень** (істинності або помилковості) і **логічними операціями** над ними.

Під висловленнями розуміються ствердження, щодо змісту яких, можна стверджувати, істинні вони або помилкові. Наприклад, висловлення "кіт - тварина" - істинно, "всі кути прямі "- помилкове.

Для позначення істинності вводиться символ "І" а помилковості - символ "П". Потім замість цих символів стали вживати числа {1,0}, як двійкові змінні, тобто логічна «1» (лог.1), логічний «0» (лог.0).

Закон виключення третього - кожне висловлення може бути істинним або помилковим (третього не дане). ***При цьому, висловлення не може бути одночасно істинним і помилковим*** (**закон протиріччя**).

Істинність висловлення встановлюється на основі аналізу його змісту і обставин, якими керуються при його тлумаченні.

Наприклад: "Київ - столиця України" - істинно, а "100<10" - помилкове. Однак, перше ствердження перестає бути істинним, коли мова йде про період, коли столицею УРСР був Харків. Друге стане ***істинним***, якщо 100 записано у двійковій системі, а 10 - у десятковій (4 < 10).

Абсолютно істинне висловлення - якщо висловлення не може бути помилковим при будь-яких умовах. Позначається - Const 1.

Наприклад: "Земля обертається".

Абсолютно помилкове висловлення - якщо висловлення не може бути істинним при будь-яких умовах. Позначається - Const 0.

Наприклад: "Земля - супутник Марса".

Абсолютно істині (помилкові) висловлення називаються ***логічними константами***.

Прийняття закону виключення третього дозволяє повністю використати в логіку висловлень математичний **апарат двозначної логіки** (1, 0).

Цікаво, що зараз математичний **апарат багатозначної** логіки (що розвивається**)** дозволяє досліджувати висловлення при декількох значеннях, наприклад, "істина", "помилковість" і "невизначеність". Застосовується при дослідженні динаміки перехідних процесів. У рамках цього курсу не розглядається.

Коли **істинність ствердження** визначається з деякою ймовір-ністю, то логіка висловлень перетворюється в **ймовірнісну логіку**.

В алгебрі висловлень розглядаються питання, пов'язані з ***утворенням складних висловлень.*** Якщо в нас є кілька простих висловлень, то за допомогою логічних зв'язувань і заперечень із них можна утворити нові складні висловлення. Звичайно, простота і складність висловлень відносні.

Наприклад: "На вулиці світить сонце", "В аудиторії йде заняття". Із цих висловлень можна утворити більше десяти складних, у тому числі і безглузді висловлення, наприклад: "Якщо в аудиторії йдуть заняття, то на вулиці світить сонце".

Особливість алгебри висловлень - допускаються тільки граматично вірні способи утворення складних висловлень і зовсім ігнорується зміст висловлення, яке вийшло. В алгебрі висловлень ***цікавляться лише істинністю або помилковістю висловлень***. Більше того: у ній ***досліджується питання про істинність складного висловлення*** залежно від істинності вхідних у нього простих висловлень і, із цього погляду, досліджуються різні логічні зв'язування.

Приклад. "На вулиці світить сонце і в аудиторії йдуть заняття - нормальний навчальний процес".

Складне висловлення може бути істинним, тільки якщо перше і друге - істині.

Одне з головних питань, що нас цікавить, укладений в одержанні щирого складного висловлення із простих за допомогою логічних зв'язувань.

Поряд із простими словесними висловленнями, приклади яких приводилися вище, стали використати ***літерні позначення змінних висловлень***, тобто такі змінні x, y, z, x_1, x_2 ... x_n, значеннями яких можуть бути символи з множини {0,1} і позначають заздалегідь задані індивідуальні висловлення.

Наприклад: X_1=1 може позначати висловлення: "Поданий сигнал керування X_1 з рівнем напруги +4,5В", що відповідає лог.1.

X_1=0 може позначати висловлення: "Поданий сигнал керування X_1 з рівнем напруги ≤0,3У", що відповідає лог.0, і так далі:

X_2-"Поданий сигнал керування X_2", …;

X_3-"Поданий сигнал керування X_3", ….

Таким чином, за допомогою двозначних змінних $x_1,x_2,... x_n$ ми переходимо до булевих змінних - основи бінарної алгебри логіки.

Булевими змінними називаються змінні, приймаючі одне із двох значень множини {0,1}.

Логічна функція (ЛФ) - функція $F(x_1,x_2,…,x_n)$, що приймає значення з множини {0,1} на будь-яких наборах логічних змінних $x_1,x_2,…,x_n$, також приймаючі значення з множини {0,1}.

1.2 Кінцевий автомат

Розглянемо деякий пристрій, наведений на рисунку 1.1. Кінцевий автомат можна представити як пристрій перетворення дискретної інформації, що має кінцеву множину вхідних X и вихідних Y бінарних каналів і тактового часу, що знаходиться в кожний дискретний момент t в одному зі станів кінцевої множини Z.

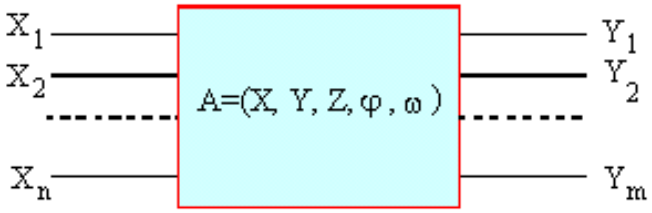


Рисунок 1.1- Кінцевий автомат

По вхідних каналах повідомлень X у кожний поточний тактовий момент t у пристрій надходить вхідний код (слово) з деякої вхідної кінцевої множини $X = \{x_1,x_2,…,x_n\}$, названого **вхідним алфавітом.**

Вказується функція переходу φ(t) стану Z(t) автомата в тактовий момент $t_{(i+1)}$, залежно від вхідного коду і стану пристрою в поточний тактовий момент t, а також функція виходу ω(t), що відбиває утворення і значення (код) вихідного слова кінцевої множини $Y = \{y_1,y_2,…,y_m\}$ **вихідного алфавіту** в такті t.

Такий пристрій, що характеризується вхідною множиною сигналів X, вихідною множиною сигналів Y, множиною внутрішніх станів Z(t) і функціями переходів φ(t) і виходів ω(t), називається кінцевим цифровим автоматом (ЦА).

ЦА без елементів пам'яті називають ***тривіальним автоматом***, що характерно для ***комбінаційних схем.***

Отже, закон функціонування кінцевого ЦА можна представити математичним кортежем:

$$\mathbf{ЦА = \langle X,Y,Z,\varphi,\omega \rangle}, \quad (1.1)$$

для якого в будь-якому такті t_i вихідне слово є функцією:

$$\mathbf{Y_i(t_i) = f_i\{X(t_i), Z(t_i), \varphi(t_i),\omega(t_i)\}}. \quad (1.2)$$

Існує два підходи при вивченні, аналізі і синтезі цифрових автоматів - макропідхід і мікропідхід. При макропідході цікавляться зовнішнім поводженням автомата. Як він здійснює переробку вхідної інформації у вихідну, у якій послідовності змінюються його стани. При цьому, не розглядається внутрішня будова автомата, зміст його логічних елементів, зв'язків. На цьому шляху приходять до поняття абстрактного кінцевого ЦА, що може бути заданий за допомогою набору вхідних повідомлень, вихідних логічних функцій, кількості його станів і переходів між ними, що описують його поводження в часі і багатомірному просторі функцій.

При мікропідході враховується состав логічних елементів, структура пристрою, міжелементні зв'язки, логічна схема і її математичний опис процесу функціонування при зміні станів $\mathbf{Z(t_i)}$ автомата в різних послідовностях, його залежність від вхідних булевих змінних та попередніх станів $\mathbf{Z(t\text{-}1)_i}$. На цьому шляху приходять до поняття структурного кінцевого ЦА. Структурний автомат може бути заданий системою логічних функцій аналітичного опису, табличним способом, графом автомата, графою-схемою алгоритму (ГСА) і інших способів. При цьому задаються або вибираються обмеження

по застосуванню кінцевої множини логічних інтегральних схем (ІС) і схемою їхньої взаємодії між собою. Узагальнення кінцевого ЦА виходить шляхом об'єднання понять абстрактного і структурного автоматів.

Відповідно до цим, вся теорія автоматів розділена на **теорію абстрактних автоматів** і **теорію структурних автоматів.**

Теорія абстрактних автоматів на рівні моделей вивчає відносини між вхідною і вихідною інформацією і станами, які відбивають поводження автомата.

Основним змістом **теорії структурних автоматів** є вивчення властивостей відображень для різних композицій автомата з кінцевої множини мікросхем щодо заданого класу логічних функцій і алгебр, аналіз і синтез мінімізованих схем.

1.3 Основні логічні операції

При утворенні із простих висловлень складних, більшу роль грають сполучні зв'язування, що визначають зміст і логіку цілого ствердження. Наприклад: "Я піду в парк ***і*** зустріну друга", "Я піду в парк ***або*** зустріну друга", "***Якщо*** я піду в парк, ***то*** зустріну друга", "Я піду в парк, ***якщо і тільки якщо*** зустріну друга", "Я піду в парк ***тоді і тільки тоді***, коли зустріну друга".

В алгебрі логіки сполучні зв'язування, крім зв'язку змінних, визначають логічні операції. Основні з них наведені нижче із прийнятими в технічній літературі позначеннями і таблицями істинності.

Таблицею істинності (ТІ) називається таблиця, у якій наведені можливі значення {0,1} вхідних змінних висловлень X ∈ (x_1, x_2,…x_n) і відповідні їм вихідні значення функції Y(x_1, x_2,…x_n) з множини {0,1}. Це є основна мета побудови складного висловлення.

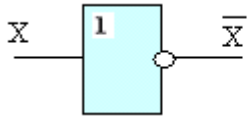
1.3.1 Операція заперечення

Операція заперечення - НІ, в алгебрі логіки позначається рискою над змінною (формулою): $\overline{X}$. Зустрічається позначення ¬X, ~X.

Запереченням називається така логічна операція між вхідною логічною змінною X і вихідною логічною змінною Y, при якій Y істинно тільки тоді, коли X помилкове, і навпаки, Y помилкове тільки тоді, коли істинно X.

За допомогою логіко-математичної символіки логічна функція НІ записується як У = $\overline{x}$ і читається "у дорівнює не х".

Наприклад, якщо X - ствердження про наявність сигналу Лог.1 на вході мікросхеми, то Y відповідає ствердженню про наявність сигналу Лог.0 (див. рис. 1.2).



X	0	1
$\overline{X}$	1	0

Рисунок 1.2 – Позначення і ТІ операції НІ

Далі для спрощення розуміння наведені приклади кількох логічних функцій (ЛФ) тільки від двох змінних Y(x_1, x_2), хоча більшість з них також вірні для будь-якої їх кількості Y(x_1, x_2,…x_n).

1.3. 2 Операція логічного множення

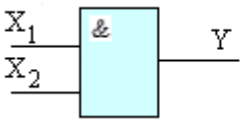
Операція логічного множення декількох змінних - І, кон`юнкція. Синоніми: збіг, добуток, перетинання, логічне І.

Логіко-математична символіка для двох змінних має кілька видів:

$y = x_1 \cdot x_2$, $y = x_1 x_2$, $y = x_1 \wedge x_2$, $y = x_1$ & x_2, $y = x_1 \cap x_2$. Читається: x_1 і x_2.

Операцією логічного множення декількох вхідних змінних X_i (x_1, x_2,.., x_n) ***називається така вихідна функція Y, що істинна тільки тоді, коли одночасно істинні всі множені змінні з X_i*** (x_1, x_2,…x_n).

Логічна операція двомісна і більше.



X_1	0	0	1	1
X_2	0	1	0	1
Y	0	0	0	1

Рисунок 1.3 – Позначення і ТІ операції І

Сигнал на виході логічного пристрою І буде тоді, коли буде x_1 = Лог.1 і x_2 = Лог.1. Це добре видно з ТІ (див. рис. 1.3).

1.3. 3 Операція логічного додавання

Операція логічного додавання АБО називається диз'юнкція (поділ, логічна сума, збірка, логічне АБО).

Логіко-математична символіка для двох змінних має кілька видів:

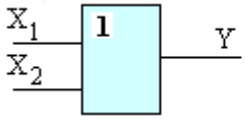
$y = x_1 + x_2$, $y = x_1 \vee x_2$, $y = x_1 \cup x_2$. Читається: x_1 або x_2.

Операцією логічного додавання кількох змінних X_i $(x_1, x_2, \ldots x_n)$ ***називається така функція Y, що істинна тоді, коли істинна хоча б одна змінна з X_i*** $(x_1, x_2, \ldots x_n)$.

Логічна операція АБО двомісна і більше. Треба мати на увазі, що у звичайній мові сполучення "або" вживається у двох різних змістах: **не альтернативне (не виключає) АБО** і альтернативне (виключне) АБО. У висловленнях першого типу затверджується істинність, принаймні, одного із простих висловлень, що беруть участь у ньому; у другому випадку - у точності одного (або ... або).

Диз'юнкція відповідає **не альтернативній операції АБО**. Абсолютна істинність $x_1 \vee x_2$ означає, що в кожній ситуації хоча б одне з висловлень x_1, x_2 істинно, а якщо обоє, те тим більше є істинно.

Сигнал на виході логічного пристрою АБО буде Лог.1 тоді, коли хоча б один сигнал на вході буде 1 (див. рис. 1.4).



X_1	0	0	1	1
X_2	0	1	0	1
Y	0	1	1	1

Рисунок 1.4 – Позначення і ТІ операції АБО

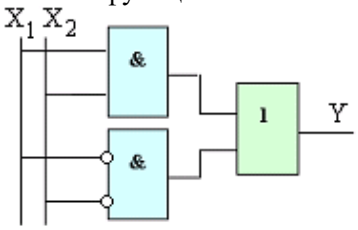
1.3. 4 Операція еквіваленція

Називається - еквіваленція (рівнозначність, еквівалентність, взаємозалежність). Логічна операція двомісна і більше. Логіко-математична символіка для двох змінних має кілька видів зображень:

$y = x_1 \sim x_2$, $y = x_1 \equiv x_2$, $y = x_1 \leftrightarrow x_2$. Читається: x_1 як x_2.

Операцією еквіваленції називається така функція декількох змінних $X_i(x_1, x_2, \ldots x_n)$ функція Y котрої істинна тоді, коли всі змінні $X_i(x_1, x_2, \ldots x_n)$ або істинні або помилкові.

Звичайно логічний елемент еквіваленції будують на елементах І, АБО, НІ (див. рис. 1.5), наприклад, на основі еквівалентності функцій.



X_1	0	0	1	1
X_2	0	1	0	1
Y	1	0	0	1

$Y = X_1 X_2 + \overline{X}_1 \overline{X}_2$

Рисунок 1.5 – Побудова і ТІ операції еквіваленції

1.3. 5 Операція імплікація

Називається імплікація (поділ із забороною, проходження, селекція). Логіко-математична символіка для двох змінних має вигляд:

$y = x_1 \rightarrow x_2$, $y = x_1 \supset x_2$ $y = x_1 > x_2$. Читається: якщо x_1 ..., то x_2.

Логічна операція **двумістна**. Просте висловлення x_1 називається посилкою імплікації, x_2 - її висновком.

У випадку імплікації невідповідність між звичайним розумінням істинності складного висловлення і точкою зору змісту висловлень ще помітніше, ніж для інших логічних операцій.

Наприклад: «Якщо в Запоріжжя йде дощ, то всі дороги брудні».

Головна відмінність полягає в тому, що значеннєвий зміст має на увазі причинний зв'язок між посилкою ***(x_1, x_2)*** і висновком ***$Y(x_1, x_2)$***.

Операцією імплікації називається така функція $Y(x_1, x_2)$, що помилкова тоді і тільки тоді, коли x_1 істинно, а x_2 помилкове.

На рис. 1.6 представлені **дві схеми** логічної функції імплікації побудованої на елементах І, АБО, НІ для двох еквівалентних функцій.

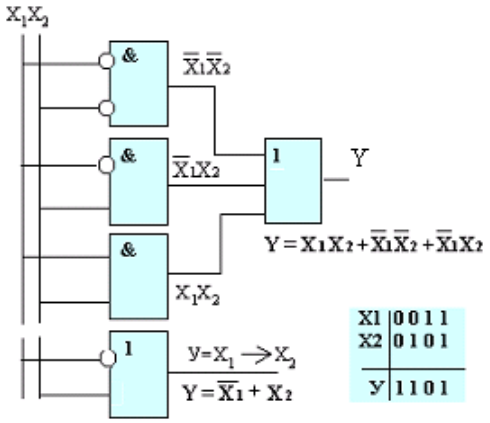


Рисунок 1.6 – Побудова і ТІ операції імплікації

1.3. 6 Сума по модулю 2

Називається: сума по модулю 2 (додавання по модулю 2, нерівнозначність, антиеквівалентність). Логіко-математична символіка двох змінних має кілька видів:

$y = x_1 \oplus x_2$, $y = x_1 \nabla x_2$, $y = x_1 + x_2$, $y = f(x_1, x_2, ..., x_n) \cdot \mathbf{mod2}$.

Читається: додавання x_1 і x_2 по модулю 2. Логічна операція двомісна і більше.

Операцією додавання змінних X_i ($x_1, x_2, \ldots, x_n$) по mod2 називається функція Y ($x_1, x_2, \ldots, x_n$), що помилкова тоді, коли число істинних змінних парне (або дорівнює нулю) і навпаки.

Для функції $y = x_1 \oplus x_2$ технічна реалізація (див. рис. 1.7) будується за допомогою наступної відповідності:

$$x_1 \oplus x_2 = x_1 \overline{x}_2 + \overline{x}_1 x_2 = (x_1 + x_2) \cdot (\overline{x}_1 + \overline{x}_2).$$

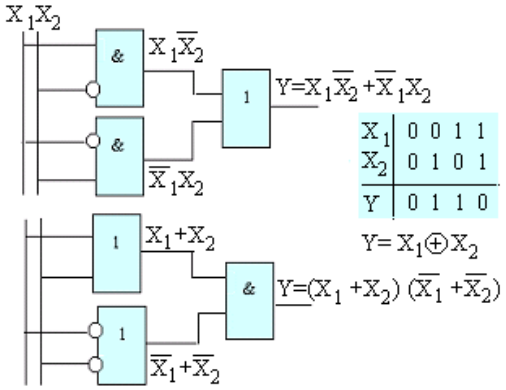


Рисунок 1.7 – Побудова і ТІ операції сума по модулю 2

Дуже часто застосовується в техніці ЦА для контролю вірності передачі інформації по інформаційному каналу.

1.3. 7 Штрих Шеффера

Називається: **штрих Шеффера** (заперечення кон`юнкції, несумісність, логічне І-НІ). Ця функція названа по імені німецького математика Шеффера, що на основі цієї функції створив свою алгебру логіки. Логічна операція двомісна і більше. Логіко-математична символіка: $y = x_1 / x_2$, $y = \overline{x_1 \wedge x_2}$, $y = \overline{x_1 \,\&\, x_2}$. Читається: x_1 і x_2 ні.

Операцією штрих Шеффера називається функція Y (x_1, x_2) яка помилкова тільки тоді, коли x_1 і x_2 істинні.

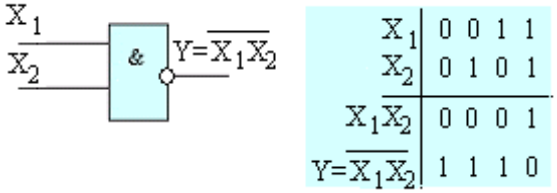


Рисунок 1.8 – Побудова і ТІ операції штрих Шеффера

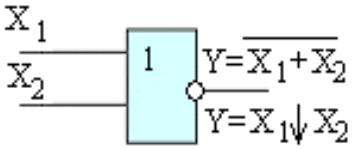
Для функції $y = x_1 / x_2$ технічна реалізація (див. рис. 1.8) будується за допомогою наступної відповідності $y = x_1 / x_2 = \overline{x_1 x_2}$.

1.3. 8 Стрілка Пірса

Називається: **стрілка Пірса** (функція Вебба, заперечення диз'юнкції, логічне АБО-НІ).

Математики Пірс і Вебб, незалежно друг від друга властивості, що вивчали, цієї функції, створили алгебру логіки, названу алгеброю Пірса. Логічна операція АБО двомісна і більше. Логіко-математична символіка: $y=x_1 \downarrow x_2$, $y=x_1 \circ x_2$, $y=\overline{x_1 \vee x_2}$. Читається: x_1 або x_2 не.

Операцією стрілка Пірса (Вебба) називається функція Y (x_1,x_2), яка істинна тільки тоді, коли помилкові всі змінні x_1 і x_2.



X_1	0	0	1	1
X_2	0	1	0	1
$Y=\overline{X_1+X_2}$	1	0	0	0

Рисунок 1.8 – Побудова і ТІ операції стрілка Пірса

Ми розглянули основні логічні операції - або ЛФ (які знайшли найбільше застосування в техніці ЦА) від однієї і двох змінних. Проведемо аналіз цих логічних функцій (ЛФ) і їхної залежності від числа вхідних змінних та розглянемо обсяг і состав ЛФ залежно від кількості вхідних змінних.

2 ЗАЛЕЖНІСТЬ СОСТАВА ФУНКЦІЙ ВІД ЧИСЛА ЗМІННИХ

2.1 Склад функцій при відсутності вхідних змінних

Представимо, що на вхід логічної схеми не надходить жоден сигнал. У цьому випадку на виході може бути або 0, або 1, тобто для числа змінних n=0 на виході може бути дві ЛФ, **$У_0$=0, $У_1$=1**. Їх називають **const 0, const 1**. Вони постійні, тому що вхідний сигнал відсутній. Константа 0 і константа 1 позначають абсолютно помилкову і абсолютно істинну функцію, що не змінюється. (На практиці ці функції мають місце при замиканні або згорянні усередині мікросхеми ЦА якого-небудь елемента або ланцюга їхнього з'єднання. У цьому випадку будь-які зміни вхідних (x_1, x_2,…x_n) не приводять до зміни сигналу Y на виході – тобто є аварійна ситуація).

В техніці таких логічних елементів немає, але їх легко одержати, наприклад, з логічного елемента І, І-НІ на дві змінні (див. рисунки 2.1).

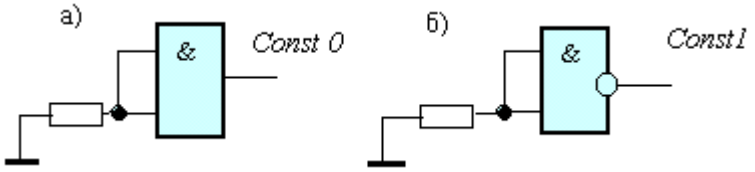


Рисунок 2.1 - Отримання абсолютних констант: *Const 0* на елементі І – а; *Const 1* на елементі І-НІ.

2.2 Функції однієї змінної

У таблиці 2.1 наведені функції для однієї змінної. Змінна X має два значення 0 і 1, тому, загальне число функцій дорівнює чотирьом, тому що із двох розрядів можна одержати чотири різні комбінації.

Таблиця 2.1- Функції однієї змінної

X	0 1	Позначення	Назва	Читання
Y_0	0 0	Const 0	Константа 0	Завжди 0
Y_1	0 1	f(x) ≡ x	Повторення	Як x
Y_2	1 0	$\overline{x}$	Заперечення	Не x
Y_3	1 1	Const 1	Константа 1	Завжди 1

Функції Y_0, Y_3 вже знайомі нам ЛФ – Const (0,1). Функція Y_2 = f(x) = x, повторює значення логічної змінної, називається ***тотожна функція.*** Функція Y_3 = $\overline{x}$ протилежна за своїм значенням логічної змінної X - ***логічне заперечення*** або функція НІ (інверсія).

2.3 Функції двох змінних

У таблиці 2.2 наведені функції **двох змінних**. Дві змінні мають чотири різних комбінації вхідних кодів (наборів). Кожному набору відповідає один розряд функції на якому вона дорівнює 1

або 0. Номер функції зручно прив'язати до послідовного бінарного рахунку. Під кожним набором виходять значення $\mathbf{Y_i}$.

Таблиця 2.2 - Булеві функції двох змінних

X_1 X_2	**0 0 1 1** **0 1 0 1**	**Позначення**	**Назва**	**Читання**
$\mathbf{Y_0}$	0 0 0 0	Const 0	Константа 0	Завжди 0
$\mathbf{Y_1}$	0 0 0 1	$x_1 x_2$	Кон`юнкція	x_1 і x_2
$\mathbf{Y_2}$	0 0 1 0	$x_1 \leftarrow x_2$	Заперечення імплікації	x_1, але не x_2
$\mathbf{Y_3}$	0 0 1 1	x_1	Повторення	Як x_1
$\mathbf{Y_4}$	0 1 0 0	$x_2 \leftarrow x_1$	Запереч.обернен.імплік.	Не x_1, але x_2
$\mathbf{Y_5}$	0 1 0 1	x_2	Повторення	Як x_2
$\mathbf{Y_6}$	0 1 1 0	$x_1 \oplus x_2$	Σ mod 2	x_1 не як x_2
$\mathbf{Y_7}$	0 1 1 1	$x_1 + x_2$	Диз'юнкція	x_1 або x_2
$\mathbf{Y_8}$	1 0 0 0	$x_1 \downarrow x_2$	Стрілка Пірса	x_1, або x_2 не
$\mathbf{Y_9}$	1 0 0 1	$x_1 \sim x_2$	Еквіваленція	x_1 як x_2
$\mathbf{Y_{10}}$	1 0 1 0	$\overline{x}_2$	Заперечення 2-го аргум.	Не x_2
$\mathbf{Y_{11}}$	1 0 1 1	$x_2 \rightarrow x_1$	Обернена імплікація	Якщо x_2, то x_1
$\mathbf{Y_{12}}$	1 1 0 0	$\overline{x}_1$	Заперечення 1-го аргум.	Не x_1
$\mathbf{Y_{13}}$	1 1 0 1	$x_1 \rightarrow x_2$	Імплікація	Якщо x_1, то x_2
$\mathbf{Y_{14}}$	1 1 1 0	x_2 / x_1	Штрих Шеффера	x_1 і x_2 не
$\mathbf{Y_{15}}$	1 1 1 1	Const 1	Константа 1	Завжди 1

Аналізують функції і привласнюють їм назви по їхньому поводженню щодо всіх чотирьох наборів змінних. Аналіз показує, що шість із наведених функцій не залежать від x_1 або x_2 (або від обох разом). Це дві константи (Y_0,Y_{15}), повторення (Y_3,Y_5), і заперечення (Y_{10},Y_{12}). З інших 10 функцій дві (Y_4,Y_{11}) відрізняються від відповідних їм (Y_2,Y_{13}) лише порядком розташування аргументів і тому не є самостійними. Тому з 16 функцій двох змінних тільки 8 є оригінальними (Y_1, Y_2, Y_6, Y_7, Y_8, Y_9, Y_{13}, Y_{14}).

Розгляд булевих функцій Y від однієї, двох і більшого числа змінних (x_1,x_2,…,x_n) показує, що ***будь-яка функція від меншого числа змінних має місце серед функцій більшого числа змінних.***

Визначення. ***Функції, які зводяться до залежності від меншого числа змінних, називаються вироджувальними.***

Функції, що істотно залежать від всіх змінних, називаються невироджувальними.

Функції рівносильні одна одній, якщо приймають на всіх можливих наборах змінних ті самі значення $f_1(x_1, x_2,\ldots, x_n) = f_2(x_1, x_2,\ldots, x_n)$.

2.4 Дійсні і фіктивні функції

Булеві змінні можуть бути **дійсними або фіктивними.**

Змінна x_i дійсна (суттєва), якщо значення функції $f_1(x_1,x_2,\ldots,x_n)$ істотно змінюється при зміні x_i.

Змінна x_i фіктивна (несуттєва), якщо значення функції $f_1(x_1,x_2,\ldots,x_n)$ не змінюється при зміні x_i.

Наприклад, у таблиці 2.2 функція $y_3 = f(x_1, x_2)$ від зміни x_2 не залежить, тобто є вироджувальною, те ж для Y_0, Y_3, Y_5, Y_{10}, Y_{12}, Y_{15}.

Визначення. ***ФАЛ не зміниться, якщо до її аргументів дописати будь-яке число фіктивних змінних або закреслити ті, які для даної функції є фіктивними.***

Правило: Для знаходження несуттєвих (фіктивних) змінних таблично заданої функції алгебри логіки (ФАЛ), виписуємо окремо підмножини f(x)=l і f(x)=0. Для перевірки фіктивності змінної x_i, викреслюємо її стовпці. Якщо при цьому в обох підмножинах не з'явилися однакові набори, значить x_i несуттєва (фіктивна).

Приклад. Нехай функція задана ТІ (див. таблицю 2.3).

Таблиця 2.3 - Таблиця істинності

X_1	X_2	X_3	$f(x_1, x_2, x_3)$	X_1	X_2	X_3	$f(x_1, x_2, x_3)$
0	0	0	0	1	0	0	1
0	0	1	0	1	0	1	1
0	1	0	1	1	1	0	0
0	1	1	1	1	1	1	0

Для визначення істотних або фіктивних функцій, проведемо розбивку наборів f(x)= l і f(x)= 0. Викреслимо в наборах другий стовпець - x_2. У нульових і одиночних стовпцях з'явилися однакові набори. Значить x_2 - суттєва. Викреслимо стовпець x_3. Однакових наборів немає. Значить x_3 - несуттєва.

Одиничні набори	Нульові набори
010	000
011	001
100	110
101	111

2.5 Визначення загального числа функцій

У кінцевому цифровому автоматі число вхідних змінних звичайно, число станів змінної x_i, звичайно, тому можна визначити загальне число різних функцій N для будь-якого числа змінних.

Теорема. Число N різних функцій алгебри логіки, що залежать від n аргументів (змінних), звичайно і дорівнює $N=2^{2^n}$. Для доказу теореми розглянемо таблицю 2.4 для n змінних. Набори змінних назвемо **кортежами**. При такому впорядковані кортежів, ліворуч є нульовий набір, праворуч - одиничний.

Таблиця 2.4 - Число всіх функцій для n змінних

X_1 X_2 X_3 … X_n	0 1 0 1 0 1 0 1...0 1 0 0 1 1 0 0 1 1...1 1 0 0 0 0 1 1 1 1...1 1 …… 0 0 0 0 0 0 0 0...1 1	Число можливих наборів 2^n
Y_0 Y_1 Y_2 Y_3 … Y_{2^n-1}	0 0 0 0 0 0 0 0...0 0 0 0 0 0 0 0 0 0...0 1 0 0 0 0 0 0 0 0...1 0 0 0 0 0 0 0 0 0...1 1 …… 1 1 1 1 1 1 1 1...1 1	Число всіх функцій $=2^{2^n}$

У нижній частині знаходяться значення функцій $\mathbf{Y_i}$ для кожного можливого набору n змінних. Загальне число таких наборів дорівнює 2^n. Задаючи той або інший конкретний двійковий набір n - змінних, ми тим самим задаємо одне з можливих {0,1} значень функції $\mathbf{Y_i}$. Але тому що функція $\mathbf{Y_i}$ задається двійковим числом (розрядність 2^n), то загальне число N різних функцій буде дорівнювати:

$\mathbf{N = 2^{2^n}}$, тобто =0, 1, 2, ..., 2^{2^n} -1. Теорема доведена.

У число ФАЛ, підраховуваних по попередній теоремі, входять як істотно залежні функції від всіх змінних, так і функції, для яких частина змінних фіктивна.

3 СУПЕРПОЗИЦІЯ ФУНКЦІЙ

Нами розглянуті одинадцять елементарних функцій: Const 0, Const 1, повторення, кон`юнкція, диз'юнкція, сума по mod2, стрілка Пірса, штрих Шеффера, еквиваленція, імплікація, заперечення. За допомогою методу ***суперпозиції елементарних функцій,*** можна одержувати більше складні функції від будь-якого числа змінних, застосовуючи різні композиції.

3.1 Методи суперпозиції

До основних методів суперпозиції відносяться:

- метод перенумерації аргументів;
- метод підстановки у функцію нових функцій замість аргументів.

До методів перенумерації аргументів можна віднести, наприклад, функцію $x_2 \rightarrow x_1$, що є суперпозицією функції імплікації $x_1 \rightarrow x_2$.

Розглянемо таблицю 2.4. Кожному і-му кортежу (комбінації X_i) можна поставити у відповідність терм T_i, що складає деяку множину G ФАЛ. Кожної функції з G зіставимо функціональний символ Y. Нехай $x_1, x_2,..., x_n$ розрахункова множина символів вхідних змінних.

Визначимо поняття терм як:

- змінна є терм;
- якщо Y функція з G і T_1, ... ,T_m - терми, то $Y(T_1, …,T_m)$ також терм (інших визначень термів немає).

Визначимо значення терма Т при значеннях змінних $X_i \in (x_1, x_2, ..., x_n)$ з множини {0,1}:

- якщо Т змінна, то значення Т збігається зі значенням цієї змінної;
- якщо $T=Y(T_1, ...,T_m)$, а значення $T_1,...,T_m$ суть $E_1, ..,E_m$ відповідно, те, значення Т є $f(E_1, ...,E_m)$, де $E_1,...,E_m$ конкретне значення змінні функції з {0,1} для конкретного кортежу змінних.

Функція G є суперпозиція інших функцій $y_1, y_2, \ldots, y_m$, якщо G представлена термом, всі функціональні символи якого є серед $y_1, y_2,\ldots, y_m$. Наприклад: а) $\bar{x}_1 \vee x_2$; б) $x_1 \cdot x_2 \vee x_1 \cdot \bar{x}_2 \vee \bar{x}_1 \cdot \bar{x}_2$.

Існує і інше еквівалентне визначення.

Функцію, отриману з функцій $y_1, y_2,\ldots, y_m$ шляхом застосування (можливо багаторазового) наступних правил:

- перенумерації аргументів;
- підстановки у функцію нових функцій замість аргументів - будемо називати суперпозицією функцій $y_1,y_2,...y_m$.

Маючи, наприклад, з таблиці для двох змінних елементарні функції: заперечення, кон`юнкції, диз'юнкції, еквівалентності і імплікації, можна скласти наступні нові функції алгебри логіки, що є суперпозиціями цих функцій:

$x_2 \to \bar{x}_1$; $x_1 \vee \bar{x}_2$; $x_1 \cdot x_2 \vee x_1$; $(x_1 \vee x_2) \to \overline{(x_1 \vee x_2) \cdot \bar{x}_1}$;

Використовуючи таблиці, що визначають елементарні функції, можна задавати будь-яку ФАЛ, що є суперпозицією цих функцій, дотримуючи при цьому пріоритет дужок {[( )]}.

Наприклад: $F(x_1,x_2,x_3) = \{[(\bar{x}_1 \sim x_3) \vee (x_1 \downarrow x_2)](x_1 \oplus x_2)\} \to x_3$.

3.2 Вираження одних елементарних функцій через інші

Аналіз таблиці функцій для однієї і двох змінних показує, що між функціями є залежності

$\mathbf{У_i = \bar{У}_{3\text{-}i}}$, де i = 0, 1,2,3; (3.1)

$\mathbf{У_i = \bar{У}_{15\text{-}i}}$, де i = 0, 1, ... ,15.

На підставі цих залежностей можна записати співвідношення, що виражають одні елементарні функції через інші. Для підтвердження рівносильності отриманих рівнянь, можна використати ТІ. Безпосередньо по формулах 3.1 установлюємо наступні рівності, які надалі будемо широко застосовувати.

Для однієї змінної:

$0 = \bar{1}$; $1 = \bar{0}$; $x = \bar{\bar{x}}$. (3.2)

Для двох змінних:

$x_1x_2 = \overline{x_1 / x_2}$; $x_1 \leftarrow x_2 = \overline{x_1 \to x_2}$; (3.3)

$x_1 \oplus x_2 = \overline{x_1 \sim x_2}$; $x_1+x_2 = \overline{x_1 \downarrow x_2}$;

$x_1 / x_2 = \overline{x_1 \cdot x_2}$; $x_1 \to x_2 = \overline{x_1 \leftarrow x_2}$; (3.4)

або: $x_1 \sim x_2 = \overline{x_1 \oplus x_2}$; $x_1 \downarrow x_2 = \overline{x_1 \vee x_2}$.

По ТІ можна довести наступні рівності, що одержали найбільше поширення. В формулах знак логічного додавання $\vee$ (множення $\wedge$) часто замінюють на + ($\cdot$): $x_1 \to x_2 = \bar{x}_1 + x_2$; $x_1 \sim x_2 = (\bar{x}_1 + x_2) \cdot (x_1 + \bar{x}_2)$.

З підстановки в 3.3 останньої рівності (еквиваленції) одержимо:

$x_1 \oplus x_2 = \overline{(\bar{x}_1 \vee x_2)(x_1 \vee \bar{x}_2)}$; (3.5)

$x_1+ x_2 = \overline{\bar{x}_1 \cdot \bar{x}_2}$; $x_1 \cdot x_2 = \overline{\bar{x}_1 \vee \bar{x}_2}$. (3.6)

Як узагальнення 3.6 одержуємо наступні формули, названі законом **де Моргана**:

$x_1+ x_2 + x_3 +\ldots+ x_n = \overline{\bar{x}_1 \cdot \bar{x}_2 \cdot \bar{x}_2 \cdot \ldots \cdot \bar{x}_n}$; (3.7)

$x_1x_2x_3 \cdot \ldots \cdot x_n = \overline{\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \ldots + \bar{x}_n}$. (3.8)

Закон де Моргана побудований на принципі подвійності.

Визначення: ***логічне додавання n-змінних дорівнює запереченню логічного множення заперечень цих змінних.***

Логічне множення n-змінних дорівнює запереченню логічного додавання заперечень цих змінних.

Для кращого засвоєння закону де Моргана важливо запам'ятати наступне: ***логічний знак між змінними змінюється, а риса загального заперечення відноситься до всіх змінних.***

$$\overline{x_1 + x_2 + \ldots + x_n} = \overline{x}_1\,\overline{x}_2 \ldots\, \overline{x}_n, \qquad \overline{x_1 x_2 \ldots x_n} = \overline{x}_1 + \overline{x}_2 + \ldots + \overline{x}_n.$$

Якщо тепер над лівою і правою частинами рівнянь поставити загальне заперечення, що не змінить знак рівності, то одержимо наслідок закону де Моргана:

$$\overline{\overline{x_1 + x_2 + \ldots + x_n}} = \overline{\overline{x}_1\,\overline{x}_2 \ldots\, \overline{x}_n}, \qquad \overline{\overline{x_1 x_2 \ldots x_n}} = \overline{\overline{x}_1 + \overline{x}_2 + \ldots + \overline{x}_n}.$$

Урахувавши подвійні заперечення, одержимо те ж визначення:

$$x_1 + x_2 + \ldots + x_n = \overline{\overline{x}_1\,\overline{x}_2 \ldots\, \overline{x}_n}, \qquad x_1 x_2 \ldots x_n = \overline{\overline{x}_1 + \overline{x}_2 + \ldots + \overline{x}_n}.$$

Це можна записати в більше скороченому виді як узагальнений закон дуальності (**закон де Моргана –Шеннона**), що виражається:

$$\overline{\Sigma x_i} = \Pi \overline{x}_i, \qquad \overline{\Pi x_i} = \Sigma \overline{x}_i,$$

$$\Sigma x_i = \overline{\Pi \overline{x}_i}, \qquad \Pi x_i = \overline{\Sigma \overline{x}_i}.$$

Закон де Моргана і його наслідок широко використовується в аналітичних перетвореннях ЛФ при їхній мінімізації.

Висновок: Операції логічного додавання ($\vee$, +) і множення ($\wedge$, $\cdot$) є взаємно ***подвійні*** (знаки $\wedge$, $\cdot$ в виразах можуть бути опущені).