ТЕОРЕТИЧНІ ВІДОМОСТІ ДО ЛАБИ №1 ТА ПРИКЛАД ЇЇ ВИКОНАННЯ

*Булеві функції* (БФ) - це функції, що приймають лише два значення 0 і 1 і залежать від аргументів, кожен із яких також може приймати два значення - 0 і 1.

Аргументи, від яких залежать БФ, також називають *булевими* (чи *двійковими*). БФ від n змінних (аргументів) будемо позначати f(x1,…,xn).

Теорема 1. Область визначення будь-якої БФ f(x1,…,xn), що залежить від n змінних, складається з 2n кортежів (упорядкованих двійкових наборів значень змінних, від яких залежить БФ).

Доведення: доведення будемо проводити методом математичної індукції по кількості елементів, від яких залежить БФ.

1. Перевіримо твердження теореми при n=1. При n=1 твердження справедливе, оскільки, по визначенню, булева змінна може приймати лише два значення – 0 і 1. З іншого боку, згідно з твердженням теореми область визначення БФ від однієї змінної складається із 21=2 кортежів.
2. Припустимо, що твердження теореми справедливо при n=k.
3. Доведемо твердження теореми при n=k+1. Розглянемо довільну БФ від k+1 аргументу. Позначимо її f(x1,…,xk+1). Цю функцію можна представити як об’єднання двох БФ з фіксованим останнім аргументом: f(x1,…,xk,0) і f(x1,…,xk,1). Оскільки ми зафіксували останній аргумент, то кожна з цих функцій є, по суті, функцією k аргументів. Згідно з припущенням індукції область визначення кожної з цих функцій складається з 2k кортежів. Значить, область визначення БФ f(x1,…,xk+1) складається з 2k+2k=2\*2k=2k+1 кортежів. Значить, твердження теореми справедливе при n=k+1 і, значить, згідно з методом мат. індукції, твердження теореми справедливе для будь-яких n.  (теорема доведена)

*Приклад*. Згідно з теоремою 1, область визначення будь-якої БФ від двох аргументів складається із 22=4 кортежів: 00, 01, 10, 11, а область визначення БФ трьох аргументів – із 23=8: 000, 001, 010, 011, 100, 101, 110, 111.

 (кінець прикладу)

*Способи завдання БФ. Табличний спосіб*. Існують два основні способи завдання БФ: табличний і аналітичний (за допомогою вираження одних БФ через інші). При *табличному способі завдання* БФ в лівій частині таблиці виписують всі кортежі із області визначення даної БФ, а в правій - значення, які приймає БФ на тому чи іншому кортежі. Такі таблиці називають *таблицями істинності* даної БФ.

Для стандартизації процедури завдання БФ за допомогою таблиць істинності серед n-мірних кортежів (n=1, 2, 3…) вводять так названу природну (її ще називають лексико-графічною) упорядкованість. З цією метою будь-який двійковий кортеж =(1, ..., n) розглядають як представлення деякого цілого невід’ємного числа 1\*20+…+n\*2n-1 в двійковій системі числення. Наприклад, кортежу (0,1,0,1,1) відповідає число 1\*20+1\*21+0\*22+1\*23+0\*24=11. Число, що відповідає даному набору (кортежу) називається *номером набора* (кортежа). *Природний* (*лексико-графічний*) *порядок* n-мірних двійкових кортежів ми отримуємо, розташувавши ці кортежі в порядку зростання (або убування) їх номерів. Першим у розташуванні наборів (у порядку зростання) буде набір, усі компоненти якого дорівнюють нулю, а останнім - усі компоненти якого дорівнюють одиниці. Легко бачити, двійковий кортеж повністю визначається своїм номером і розмірністю, тому, якщо розмірність кортежів задана, то їх можна ототожнювати з їхніми номерами. Кортеж, що має номер і, будемо називати *і-тим* кортежем.

Дві БФ вважаються *різними*, якщо вони відрізняються значенням хоча б на одному кортежі зі своєї області визначення.

Теорема 2. Існує різних БФ від n аргументів.

Доведення є очевидним наслідком теореми 1 (провести самостійно). 

*БФ однієї та двох змінних*. Згідно з теоремою 2 існує =4 різних БФ одного аргументу. Зведемо їх в одну таблицю:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | f0(x) | f1(x) | f2(x) | f3(x) |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |

Очевидно, що f0 (x)=0, f3(x)=1, f1(x)=x. Єдиною нетривіальною функцією одного аргументу є функція f2(x). Її називають *інверсією* (або *запереченням*) змінної x і позначаютьх.

Згідно з теоремою 2 число різних БФ двох змінних дорівнює=16. Випишемо зведену таблицю всіх цих функцій:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| х | У | f0(х,у) | f1 | f2 | f3 | f4 | f5 | f6 | f7 | f8 | f9 | f10 | f11 | f12 | f13 | f14 | f15 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

Відзначимо, БФ пронумеровані таким чином, що номер функції, записаний в двійковій системі числення, дає послідовність значень відповідної функції. Наприклад, двійковому запису 1310=11012 відповідає функція f13(x,y), яка приймає наступні значення: f13(0,0)=1, f13(0,1)=1, f13(1,0)=0, f13(1,1)=1.

Деякі з наведених в таблиці БФ мають *спеціальну назву* і *позначення*. Приведемо їх, згрупувавши БФ в пари по наступному принципу: кожна БФ пари є запереченням іншої БФ цієї пари.

1) f0(x,y)=0, f15(x,y)=1. Тотожний нуль і тотожна одиниця.

2) БФ f1(x,y) називається *кон’юнкцією* і позначається одним з наступних способів: f1(x,y)=x\*y=xy=x&y=xy. Запереченням f1(x,y) є f14(x,y), яку називають *штрихом Шеффера* і позначають як f14(x,y)=.

3) БФ f7(x,y) називається *диз’юнкцією* і позначаєтьсяху. Запереченням f7 є f8(x,y), яку називають *стрілкою Пірса* (або *функцією Вебба*) і позначається f8(x,y)=xy=

4) f13(x,y) називається *імплікацією* і позначається f13(x,y) = xy. Запереченням f12(x,y) є f2(x,y) = , яка спеціальної назви не має.

5) БФf11(x,y) називається *антиімплікацією* і позначається f11(x,y)= ух, її заперечення - f4(x,y)=  спеціальної назви не має.

6) БФ f9(x,y) називається *еквівалентністю* і позначається ху. Її заперечення - f6(x,y), яка називається *додавання по модулю 2*, f6(x,y)=ху.

7) f3(x,y)=x, f12(x,y)=

8) f5(x,y)=y, f10(x,y)=

В якості прикладу вербального, тобто мовного, способу завдання БФ наведемо визначення кон’юнкції: кон’юнкція – це БФ двох аргументів, яка приймає значення одиниці тоді і тільки тоді, коли обидва аргументи приймають значення одиниці. Ще одне вербальне, еквівалентне попередньому, визначення кон’юнкції: кон’юнкція – це БФ двох аргументів, яка приймає значення нуля тоді і тільки тоді, коли хоча б один аргумент приймає значення нуля.

1. Аналітичний спосіб завдання БФ.

2. Операція суперпозиції (підстановки) БФ.

3. Операції на множині БФ.

4. Булева алгебра (БА). Спосіб утворення виразів в БА. Порядок виконання операцій в БА. Тотожність виразів в БА.

5. Основні та похідні тотожності БА та способи їх доведення. Узагальнення тотожностей БА.

Відзначимо, що з ростом числа аргументів відбувається досить швидке зростання числа БФ. Дійсно, існує = 28 =256 різних БФ трьох аргументів. Число різних БФ від 5 змінних уже перевищує чотири мільйона. Швидке зростання числа БФ з ростом числа аргументів фактично виключає можливість їх безпосереднього представлення за допомогою таблиць.

*Аналітичне представлення БФ*. Відзначимо, що фундаментальне значення БФ однієї та двох змінних в теорії БФ полягає в тому, що з них може бути побудована будь-яка БФ (від будь-якого числа змінних). Засобом для такої побудови є *суперпозиція* БФ чи, що теж саме, *підстановка* одних БФ замість аргументів в інші БФ. Можливість такої підстановки обумовлена тим, що, в силу визначення, області визначення БФ і області значень їхніх аргументів співпадають між собою.

Використовуючи суперпозицію, ми можемо аргументами кожної з розглянутих вище БФ однієї і двох змінних вважати довільні БФ. Тим самим БФ однієї і двох змінних можна розглядати не як функції, а як *операції* на множині всіх БФ. За допомогою цих операцій можуть бути побудовані різноманітні алгебри БФ.

*Булевою алгеброю* (БА)називається множина БФ, над якою визначені операції {, , }.

В БА будемо розглядати різноманітні *вирази*, що утворюються за допомогою операцій алгебри із констант 0 та 1 та змінних, під який ми будемо розуміти довільні БФ (зокрема, звичайні двійкові змінні). Порядок виконання операцій в БА: заперечення, кон’юнкція, диз’юнкція. Наявність у виразі дужок змінює звичайний порядок виконання операцій (спочатку виконуються операції в дужках).

Очевидно, будь-який вираз в названих алгебрах представляє собою деяку БФ і, в свою чергу, може бути позначений однією буквою.

Два вирази в БА називають *тотожніми,* якщо при підстановці в дані вирази конкретних двійкових наборів, представлені цими виразами БФ приймають однакові значення.

*Основні тотожності БА*. Відзначимо, що засобом для встановлення справедливості основних тотожностей служить перевірка рівності виразів, що стоять у лівій та правій частині тотожності, на всіх наборах двійкових значень змінних, що входять в дані вирази :

1)=*x* – *правило подвійного заперечення*;

2) xy=yx - *закон комутативності* для диз'юнкції;

3) x\*y=y\*x - *закон комутативності* для кон’юнкції;

4) x(yz)=(xy)z=xyz - *закон асоціативності* для “”;

5) x\*(y\*z)=(x\*y)\*z=x\*y\*z - *закон асоціативності* для “\*”;

6) x\*(yz)=x\*yx\* z - *перший дистрибутивний закон*;

7) xy\*z =(xy)(xz) - *другий дистрибутивний закон*;

8)  = \*- *правило де Моргана* для “”;

9) = - *правило де Моргана* для “\*”;

10) = 1, =0

11) x\*1 = x, x\*0 = 0

12) x0=x, x1=1

10-12 – *тотожності для констант*;

13) x = 1

14) x = 0

Із визначених основних тотожностей БА можуть бути виведені нові тотожності, які називають *похідними*. Відзначимо: що розходження між основними тотожностями і похідними є дуже умовними. Для того щоб підкреслити, що ми маємо справу не з основними, а з похідними тотожностями, замість маленьких будемо вживати в них великі латинські букви. Хоча, як і у випадку основних, так і у випадку похідних тотожностей, можна вважати, що, по-перше, букви, з яких вони складаються, є довільними виразами БА і, по-друге, будь-яку похідну тотожність можна доводити як і основні тотожності, тобто використовуючи визначення тотожності виразів у БА.

В якості прикладу розглянемо наступні похідні тотожності:

15) AA\*B=A - *закон поглинання* для диз'юнкції;

16) A\*(AB)=A - *закон поглинання* для кон’юнкції.

Тотожність 15 можна довести за допомогою наступного ланцюжку рівностей:

AA\*B=11A\*1A\*B =6A\*(1\*B)=2A\*(B1)=12A\*1=11A

Тут і далі над знаком “=” проставлено номер основної тотожності, що застосовувалась на тому чи іншому кроці доведення. 

Доказ тотожності 16:

A\*(AB)=12(A0)\*(AB)=7A0\*B =3AB\*0=11A0=12A 

Відзначимо, тотожності 15 та 16 справедливі при будь-яких значеннях А та B, зокрема тотожність 15 справедлива при B=1, а 16 - при B=0. Підставивши названі величини у відповідні тотожності, отримаємо наступні закони *ідемпотентності*:

17) AA=A

18) A\*A=A

Має місце також наступна тотожність:

19) A\*B=AB

Доведення : A\*B =7(A)\*(AB) =1(AB)=(AB)\*1=AB 

Наведені основні та похідні тотожності доведення допускають *узагальнення* на випадок будь-якої кількості змінних, від яких залежать ліві і праві частини цих тотожностей, зокрема за допомогою математичної індукції із тотожностей 11, 12, 17, 18 легко виводиться наступні твердження:

1. Якщо викинути із довільної БФ диз’юнктивні члени, що дорівнюють 0, ми не змінимо величини цієї диз’юнкції.
2. Якщо в диз'юнкції хоча б один із членів дорівнює 1, то вся функція дорівнює 1.
3. Якщо викинути із довільної кон’юнкції всі члени, що дорівнюють 1, то ми не змінимо величини цієї кон’юнкції.
4. Якщо в кон’юнкції хоча б один із членів дорівнює 0, то і вся кон’юнкція дорівнює 0.
5. Диз’юнкція або кон’юнкція будь-якого числа однакових членів А, дорівнює А.

6. Узагальнення правил де Моргана (8, 9) приводить до наступного твердження:

20) = 1 \*2 \* ... m

21) = 1 2 … m

7. Якщо до деякого виразу БА А застосована більше ніж одна операція заперечення, то не змінюючи значень, що приймає вираз А, можна виключити будь-яке парне число цих операцій.

1. Безінверсна форма виразів в БА.

2. Елементарні диз’юнкції та кон’юнкції.

3. Диз’юнктивні нормальні форми (ДНФ) БФ.

4. Кон’юнктивні нормальні форми (КНФ) БФ.

5. Методи знаходження ДНФ і КНФ довільного виразу БА.

Будь-який вираз БА А=А(х1…xn) за допомогою тотожних співвідношень, які існують в цій алгебрі, може бути приведений до виразу В, що побудований із змінних х1…xn та їх заперечень за допомогою одних лише кон’юнкцій та диз’юнкцій. Вираз В називають *безінверсною формою* виразу А. Тепер твердження 8 можна переформулювати в такий спосіб. Будь-яке вираз БА може бути приведений до безінверсної форми.

*Канонічні форми представлення виразів в БА*. *Елементарна диз'юнкція* (*кон’юнкція*) - це вираз, що представляє собою диз'юнкцію (кон’юнкцію) кінцевого числа попарно різних булевіх змінних або їх заперечень. По визначенню, до числа елементарних диз'юнкцій (конъюнкций) відносять також вирази, що складаються із однієї змінної (з запереченням або без нього), а також константи 0 (1).

*Приклад*: Елементарними диз'юнкціями є наступні вирази: 0, х, , , , а елементарними кон'юнкціями є вирази: 1, x, , xy, ,. В той же час вирази x1x2x1, ab, , y\*y, - і не елементарні диз'юнкції, і не елементарні кон’юнкції. 

Елементарні диз'юнкції (кон’юнкції), які відрізняються лише порядком розташуванням змінних, ми в подальшому розрізняти не будемо (у силу комутативності диз'юнкції та кон’юнкції).

*Диз’юнктивною нормальною формою (ДНФ)* називається диз'юнкція кінцевого числа попарно різних елементарних кон’юнкцій.

*Приклад*. ДНФ: f(x1, x2, x3) = x1 2 x2x3  x3 

*Кон’юнктивною нормальною формою (КНФ)* називається кон’юнкція кінцевого числа попарно різних елементарних диз'юнкцій.

*Приклад.* КНФ: f(x1, x2, x3)=(x12)(23)3  

Теорема 3. Існує єдиний конструктивний метод (алгоритм), який дозволяє для будь-якого виразу БА знайти рівні йому ДНФ і КНФ.

Доведення. Необхідний алгоритм може бути побудований наступним чином. Припустимо спочатку, що нам необхідно привести вираз до ДНФ. Тоді

1. Якщо даний вираз вже є ДНФ, то нічого робити не треба, інакше 2.

2. Приводимо вхідний вираз до безінверсного виду.

3. В отриманому виразі у відповідності з першим дистрибутивним законом розкриваємо всі дужки.

4. В отриманому виразі виключаємо всі 0 (на основі тотожності x0 =x) і об’єднуємо рівні диз’юнктивні члени (на основі тотожності xx=x).

Приведення до КНФ здійснюється згідно з наступним алгоритмом:

1. Якщо даний вираз є КНФ, то нічого робити не треба, інакше 2
2. приводимо даний вираз до безінверсного виду.
3. До отриманого виразу послідовно застосовуємо другий дистрибутивний закон.
4. Виключаємо “1” (на основі тотожності x\*1=x) і об’єднуємо ріні кон’юнктивні члени (на основі тотожності x\*x=x) . 

*Приклад*. Приведемо до ДНФ вираз:

f(x,y,z,*w*)=(xyz)=(xyz)()=(xyz)(x)=xyxzxyz

= xyxzz

Приведемо тепер цей же вираз до КНФ:

f(x,y,z,w)=(xyz)=(xyz)()=(xyz)()=(xy)(xyz)(x)= =(y)(x)(zx)(zy)(x)=(x)(xz)(yz)(x). 

1. Констітуентами нуля (одиниці) та їх властивості.

2. Досконалі ДНФ (ДДНФ) і досконалі КНФ (ДКНФ).

3.ДДНФ та ДКНФ заданої БФ.

4. Теорема про розклад БФ по змінній. Залишкові функції.

В подальшому будемо припускати, що зафіксована деяка множина М={х1, …, хn} булевих змінних. Як правило, в якості такої множини ми будемо розглядати множину аргументів даної БФ.

*Констітуентами* нуля (одиниці) називається елементарні диз'юнкції (кон’юнкції), що містять в своєму складі в прямому або інверсному вигляді всі змінні із даної множини М.

Для множини М={x1,…,xn} довільну констітуенту 0 можна представити у вигляді 1  2 …n, а довільну констітуенту 1 – у вигляді 1  2 …n, де і дорівнює або хі, або , і=1, 2, ..., n.

Теорема 4 (визначна властивість констітуент 1). Для будь-якої констітуенти 1 К = 1  2 …n , існує один і тільки один набір =(1,…,n) значень змінних, що входять в К, на якому ця констітуента приймає значення 1.

Дійсно, в силу твердження (4), для побудови необхідного набору потрібно вибрати значення і де і= рівними 1, якщо i= хі, і рівними 0, якщо і=і. Цією умовою кожна компонента і набору , а значить і весь набір  визначається однозначно. 

Приклад. Нехай M={x1, x2, x3}. Констітуентами 1 для цієї множини будуть, зокрема, наступні елементарні кон’юнкції:

x1\*2 \*x3 ; 1\*x2\*x3 ; x1\*x2\*3.

Згідно з попередньою теоремою, перша констітуента 1 приймає значення 1 на наборі (1,0,1), друга – на наборі (0,1,1), третя - на (1,1,0). 

Побудований в попередній теоремі набор  і констітуента одиниці K, яка на цьому наборі приймає значення “1”, називаються *відповідними один одному*.

Теорема 5 (визначна властивість констітуент 0). Для будь-якої констітуенти 0 R= 1  2 … n існує один і тільки один відповідний цій констітуенті набір =(1,…,n) значень змінних x1, …,xn , на якому ця констітуента приймає значення 0.

Дійсно, в силу твердження (2), необхідно вибрати значення i i = , рівним 0, якщо і= хі , і рівним 1, якщо і= . 

Приклад. Нехай M={x1, x2, x3}. Розглянемо дві констітуенти 0 для M: 1x23 і x123. Згідно з попередньою теоремою вони приймають значення 0 на наступних наборах: 1x23 – на наборі (1,0,1), а x123 – на наборі (0,1,1). 

*Довершеною ДНФ (ДДНФ)* називається така ДНФ, у якої всі елементарні кон’юнкції, із яких вона складається, є констітуентами 1 для заданої множини змінних М.

Приклад. Для M={ x1, x2, x3} вираз x12x31x2x3x1x23 є ДДНФ, в той час як вираз x12x3  x23  1x2 x3  є ДНФ, але не ДДНФ.

*Довершеною КНФ (ДКНФ)* називається КНФ, у якої всі елементарні диз’юнкції, із яких вона складається, є констітуентами 0.

Приклад. Нехай M={x1, x2, x3}. Тоді довершеною КНФ буде вираз (x123)( 1x2 3)( 12x3). Однак, вираз (x12)(x23)( 12x3) є КНФ, але не довершена КНФ. 

ДДНФ (ДКНФ) називають *ДДНФ (ДКНФ) деякої БФ* f=f(x1…xn), якщо, по-перше, ДДНФ (ДКНФ) дорівнює цій БФ (тобто приймає ті самі значення на тих самих наборах значень змінних) і, по-друге, множина змінних, від яких залежить ДДНФ (ДКНФ), співпадає з множиною аргументів БФ.

Нехай f=f (x1,…,xn) – довільна БФ. Позначимо через fi значення цієї функції на i-тому наборі значень змінних, через Кi – констітуенту 1, а через Ri - констітуенту 0, які відповідають цьому набору, і=. З метою скорочення записів будемо також позначати: x1x2…xn=xі; x1\*x2\*…\*xn=xi.

Теорема 6 (про розклад БФ по змінній). Для довільної БФ справедливі наступні формули, які називаються *формулами розкладу цієї БФ по змінній* :

; .

Доведемо справедливість формули а). Покажемо, що функції, які стоять в обох частинах рівності а) при однакових значеннях їх аргументів приймають однакові значення. Розглянемо спочатку всі можливі набори значень аргументів виду , де . На цьому наборі рівність а) приймає вигляд:

Розглянемо далі всі можливі набори значень аргументів виду , де . У цьому випадку

Таким чином, БФ в обох частинах рівності а) приймають однакові значення при будь-яких однакових значеннях їх аргументів і, значить, рівність а) справедлива.

Аналогічно доводиться рівність b) (провести самостійно). 

БФ і називаються *залишковими функціями* для функції при її розкладі по .

Відзначимо, що подібним чином можуть бути записані формули розкладу БФ по будь-якій її змінній.

Приклад. Розглянемо результат розкладу БФ по змінним і



Наслідок. Якщо розкласти задану БФ по всім змінним, то в результаті отримуємо ДДНФ (ДКНФ) даної БФ.

Дійсно, якщо розкласти задану БФ по всім змінним, то в результаті отримуємо: =fiKi =Kj. Вираз, що стоїть у правій частині, очевидно, є ДДНФ.

Доведення для ДКНФ проводиться аналогічно 

1. Теорема про існування і єдність представлення заданої БФ у вигляді ДДНФ і ДКНФ.

2. Для заданої БФ, що задана аналітично, знаходження рівної їй ДДНФ і ДКНФ. Приклади.

3. Для заданої БФ, що задана таблично, знаходження рівної їй ДДНФ.

4. Для заданої БФ, що задана таблично, знаходження рівної їй ДКНФ.

Очевидно, у даної БФ існує необмежена кількість як ДНФ, так і КНФ, що дорівнюють їй. Одні з них більш громіздкі і складні, інші - більш прості.

В той же час справедлива наступна:

Теорема 7. Для будь-якої БФ f(x1,…,xn) існує одна і тільки одна рівна їй ДДНФ і ДКНФ.

Доведення. Доведемо твердження теореми для ДДНФ.

1) Існування ДДНФ для даної БФ випливає із наслідку 2 попередньої теореми.

2) Єдиність представлення БФ у вигляді ДДНФ можна довести, наприклад, таким чином.

Припустимо, що деяка БФ f(x1,…,xn) має два (різних) представлення у вигляді ДДНФ. Розглянемо довільну констітуенту одиниці, яка входить в одне представлення і не входить в інше і розглянемо відповідний цій констітуенті набір значень змінних, від яких вона залежить. Обчислимо значення, які приймають обидва представлення даної БФ у вигляді ДДНФ. Очевидно, що одне представлення (яке містить в собі констітуенту 1, що розглядається) прийме на цьому наборі значення 1, а інше - 0. Ми тримали протиріччя (оскільки одна і та ж сама функція на одному і тому ж самому наборі приймає різні значення), яке і доводить теорему.

Твердження теореми для ДКНФ доводиться аналогічно. 

Теорема 8. Існує алгоритм, який дозволяє для будь-якого виразу БА знайти рівні йому ДДНФ і ДКНФ.

Доведення. Потрібний алгоритм має вигляд:

1. Приводимо довільне вираз БА до ДНФ (КНФ) (див. теорему 3).
2. До тих пір, доки не всі елементарні кон’юнкції (диз’юнкції), що входять в ДНФ (КНФ), не є констітуентами 1 (0):
3. Вибираємо елементарну кон’юнкцію p (диз’юнкцію q), що входить в ДНФ (КНФ), яка не містить в собі деякої із змінних хi i=, від яких залежить дана БФ.
4. В ДНФ (КНФ) замінюємо кон’юнкцію p (диз’юнкцію g) рівним йому в силу тотожностей 11 і 13 (12 і 14) виразом: p=р(xii)=pxip (g=gxi= =(gxi)(gi)) 

Приклад. Вираз f(x, y, z) = (x  z) привести до ДДНФ і ДКНФ.

а) Приведемо спочатку до ДДНФ:

f(x, y, z)= (x  z)=()(x  z) = (  )(x  z) =

=()(x  z) ( z) =  (y  )z =  y z  z

б) Приведення до ДКНФ:

f(x, y, z)= (x  z)=()(x  z) = (  )(x  z) =

=(  y)(x  z)(x  y  z) =

=( y )(  )(x   z )(   z)(x  y  z )(x v v z). 

Алгоритм знаходження ДДНФ (ДКНФ) для БФ, що задані таблицею, полягає в наступному:

* 1. Вибираємо із таблиці ті набори значень двійкових змінних, на яких дана БФ приймає значення 1 (0).
  2. Записуємо відповідні цим наборам констітуенти 1 (0).
  3. Об’єднуємо отримані констітуенти 1 (0) знаками диз’юнкції (кон’юнкції).

Приклад. Нехай БФ задана наступною таблицею істинності:

|  |  |  |  |
| --- | --- | --- | --- |
| х | y | z | F(х, y, z) |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

1) Знайдемо рівну цій БФ ДДНФ:

f(x,y,z)= yz x xy xyz

2) Знайдемо рівну цій БФ ДКНФ:

f(x,y,z)=(xyz)(xy)(y)



1. Канонічна задача мінімізації БФ.

2. Імпліканта, проста імпліканта. Проста імпліканта, що накриває одиницю в таблиці істинності БФ.

3. Повна і приведена система простих імплікант.

4. Скорочена і тупикова ДНФ. Мінімальна ДНФ (МДНФ).

5. Теорема Квайна про зв’язок тупикової і МДНФ.

6. Наслідок теореми Квайна: основні етапи вирішення канонічної задачі мінімізації БФ.

Спочатку будемо вирішувати так звану *канонічну задачу мінімізації БФ*, що полягає в знаходженні мінімального представлення даної БФ в класі ДНФ (КНФ).

*Імплікантою* БФ f(x1, . . . ., xn) називається БФ g(x1,…,xn), якщо для будь-якого набору  двійкових значень змінних, на якому БФ g приймає значення 1, випливає, що f на цьому наборі також приймає значення 1, або, іншими словами, g імпліканта f, якщо g()=1  f()=1.

Приклад. Для даної БФ, представленої в вигляді ДДНФ або ДНФ, імплікантами будуть, очевидно, констітуенти одиниці і/чи елементарні кон’юнкції, із яких дана БФ складається.

*Простою імплікантою* називається елементарна кон’юнкція, яка, по-перше, є імплікантою даної БФ, і, по-друге, ніяка власна частина цієї елементарної кон’юнкції імплікантою вже не являється.

Припустимо, що на деякому кортежі  f()=1. Будемо говорити, що ця одиниця *накривається* імплікантою g(x1,…,xn), якщо на цьому наборі g()=1

Система простих імплікант БФ (f) називається *повною* якщо будь-яка 1 із таблиці істинності даної БФ (f) накривається хоча б однією простою iмплікантою даної системи .

Система простих імплікант називається *приведеною*, якщо, по-перше, вона є повною, і, по-друге, ніяка власна частина цієї системи вже не є повною.

*Скороченою* ДНФ називається ДНФ, що складається з диз’юнкції простих інплікант, які входять в повну систему.

*Тупиковою* ДНФ називається ДНФ, що складається з диз’юнкції простих інплікант, які входять в приведену систему.

*Мінімальною* ДНФ (МДНФ) називають ДНФ, що складається з мінімальної кількості змінних, від яких залежить дана БФ, при цьому входження однієї і тієї ж самої змінної (в прямому або інверсному вигляді) в різні елементарні кон’юнкції (прості імпліканти) вважаються різними входженнями цієї змінної.

Приклад. МДНФ f (x,y,z) = xy xz yz складається з шести букв. 

Теорема (Квайна). МДНФ БФ f є тупиковою ДНФ цієї БФ або, іншими словами, якщо ДНФ не тупикова то вона не мінімальна.

Доведення. Представимо МДНФ f у виді f=рQ де р – одна із елементарних кон’юнкції, з яких складається МДНФ, а Q – диз’юнкція всіх інших елементарних кон’юнкцій.

Припустимо, що елементарна кон’юнкція р не є простою імплікантою БФ f. Тоді існує елементарна кон’юнкція r, що складається з частини змінних р, яка також є імплікантою БФ f. У відповідності з властивостями кон’юнкції, іимпліканта r=1 на всіх наборах, на яких дорівнює 1 імпліканта р. Тоді, очевидно, система імплікант r і Q є повною системою БФ f і f=rQ. Вираз rQ є ДНФ БФ f, при чому ДНФ, яка містить менше число букв, ніж початкова МДНФ рQ.

Ми отримали протиріччя, джерелом якого є припущенням про те, що імпліканта р – непроста. Таким чином, імпліканта р – проста і, в силу довільності вибору р, ми приходимо до висновку, що МДНФ будь-якої БФ f представляє собою диз’юнкцію деякого числа її простих імплікант. Із мінімальності ДНФ випливає також, що система простих імплікант є обов’язково приведеною. 

Із теореми Квайна випливає *загальна схема* рішення канонічної задачі мінімізації БФ, яку умовно можна поділити на *три етапи*:

1) знаходять всі прості імпліканти заданої БФ або, що теж саме, її скорочену ДНФ;

2) знаходять приведені системи простих імплікант і будуються тупикові ДНФ;

3) із числа тупикових ДНФ вибирають мінімальну ДНФ.

1. Деякі методи вирішення задач другого етапу канонічної задачі мінімізації БФ: метод перебору всіх варіантів.

2. Імплікантна таблиця.

3. Ядро БФ та його властивості.

2. Метод Петрика вирішення задач другого етапу канонічної задачі мінімізації БФ.

Розглянемо спочатку деякі методи рішення задач другого етапу мінімізації БФ.

Основним апаратом для рішення задач другого етапу є *імплікантна таблиця* (ІТ) БФ. Вона представляє собою прямокутну таблицю, строчки якої позначаються різними простими імплікантами БФ f, які ми знаходимо на першому етапі мінімізації БФ, а стовбці – наборами значень змінних (або відповідними їм констітуентами 1), на яких БФ приймає значення 1. Якщо деяка проста імпліканта р приймає значення 1 на наборі , що позначає деякий стовпчик ІТ, то на перетинанні р-ої строчки і -го стовпчика ставиться “\*“. Якщо імпліканта Р на наборі  дорівнює 0, то перетин р-ої строчки і -го стовпчика в ІТ залишається пустим.

При позначенні стовпчиків ІТ констітуентами 1 має силу наступне *правило заповнення* ІТ: на перетині р-ої строчки і К-го стовпчика ІТ тоді і тільки тоді ставиться “”, коли імпліканта р складає деяку власну частину констітуенти К (що, можливо, співпадає з усією констітуентою).

Приклад. Побудувати ІТ для наступної БФ: f=y  x xz yz.

Очевидно, ця БФ має чотири прості імпліканти y, x, xz, yz і приймає значення 1 на 5-ти наборах: (0,1,0); (0,1,1); (1,0,0); (1,0,1); (1,1,1). Цим наборам відповідають констітуенти 1 y; yz; x; xz; xyz.

Відповідно до приведеного вище правила, ІТ БФ f буде мати вигляд:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | y | yz | x | xz | xyz |
| y | \* | \* |  |  |  |
| x |  |  | \* | \* |  |
| xz |  |  |  | \* | \* |
| yz |  | \* |  |  | \* |

Визначення приведених систем простих імплікант може бути проведено безпосередньо по ІТ. З цією метою *потрібно вибрати* *мінімальну систему строчок таблиці* таким чином, щоб для кожного стовпчика ІТ серед вибраних строчок знайшлася хоча б одна строчка, яка містить в цьому стовпчику “\*”. Очевидно, що в будь-яку таку систему повинні входити всі строчки, в яких міститься “\*” , що є єдиною“\*” в своєму стовпчику. Прості імпліканти, що позначають строчки із указаною властивістю, складають так зване *ядро БФ*.

Легко бачити, що *прості імпліканти з ядра входять у будь-яку приведену систему будь-яких імплікант цієї функції*. Для нашого прикладу ядро БФ складають прості імпліканти y і x (їм відповідають “однозіркові” стовпчики y, x). Прості імпліканти, які входять в ядро, накривають лише частину одиниць БФ. По відношенню до ІТ більш зручно говорити, що прості імпліканти із ядра накривають відповідні цим одиницям констітуенти одиниці БФ. Очевидно, що накритими будуть всі ті і тільки ті констітуенти, які позначають стовпчики ІТ, що містять “\*” в строчках, позначення яких належать ядру. Для нашого прикладу це будуть констітуенти: y, yz, x, xz.

Якщо виключити із ІТ стовпчики, що позначені вже накритими констітуентами, ми *методом перебору* можемо знайти мінімальні покриття констітуент, що залишилися. Для нашого прикладу єдина констітуента, що залишилася не накритою - це xyz, може бути накрита як імплікантою xz, так і імплікантою yz. У відповідності з цим ми отримаємо дві приведені системи простих імплікант: {y, x, xz} і {y, x, yz}. Їм відповідають дві тупикові ДНФ: f1=yxxz і f2=yxyz із яких вибираємо МДНФ. 

Відзначимо, наведений вище метод перебору всіх можливих мінімальних покриттів констітуент безпосередньо по ІТ має практичне застосування лише у випадку відносно простих ІТ. Він може також застосовуватись в тому випадку, коли потрібно знайти не всі, а лише одне із мінімальних покрить. Для знаходження всіх мінімальних покриттів у випадку складних ІТ, як правило, застосовують алгебраїчний метод, який запропонував Петрик.

Метод Петрика. Суть цього методу полягає в тому, що по ІТ БФ f(x1…xn) будується деякий вираз, який називається *кон’юктивним представленням ІТ* (КПІТ). З цією метою позначають строчки ІТ (всі прості імпліканти БФ f) різними буквами:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | y | yz | x | xz | хyz |
| A y | \* | \* |  |  |  |
| B x |  |  | \* | \* |  |
| C xz |  |  |  | \* | \* |
| D yz |  | \* |  |  | \* |

Після цього для кожного стовпчика λ будується диз’юнкція gλ =A1…Ai всіх букв, що позначають строчки ІТ, на перетині яких зі стовпчиком λ стоять зірочки. Утворивши кон’юнкцію побудованих диз’юнкцій gλ  для всіх стовпчиків λ ІТ, ми приходимо до КПІТ. Для нашого прикладу КПІТ буде мати вигляд: φ=A(AD)B(BC)(CD)

Якщо в КПІТ у відповідності з першим дистрибутивним законом розкрити всі дужки, то виникне деякий вираз, який називається *диз’юктивним представленням ІТ* (ДПІТ). Якщо в ДПІТ виконати всі можливі операції поглинання і ідемпотентності, то ми отримаємо так зване *приведене ДПІТ*. *Суть метода Петрика* полягає в тому, що елементарним кон’юнкціям приведеного ДПІТ відповідають всі приведені системи простих імплікант даної БФ. Для нашого прикладу маємо:

φ=A(AD)B(BC)=(AAD)(BBC)(CD)=AB(CD)=ABCABD.

Таким чином, у повній відповідності з тим, що було отримано раніше, ми знайшли дві приведені системи простих імплікант: ABC ={y, x, xz} і ABD={y, x, yz}, яким відповідають дві тупикових ДНФ: f1=y  x xz і f2=y  x yz. Із f1 і f2 вибираємо МДНФ. 

1. Деякі методи вирішення задач першого етапу канонічної задачі мінімізації БФ: метод Квайна. Операція склеювання, операція неповного склеювання.

2. Метод Блейка. Операція узагальненого склеювання.

3. Метод Нельсона.

Деякі методи вирішення задач першого етапу мінімізації БФ

Метод Квайна. В процесі мінімізації БФ дуже часто використовується *операція склеювання*: AxA=A (її легко довести, дійсно, AxA=A (x)=A\*1=A) 

Якщо до виразів Ax і A була застосована операція склеювання, то говорять, що вирази Ax і A *склеюються по змінній* x.

Метод Квайна знаходження скороченої днф базується *на операції неповного склеювання*: AxA=AAxA

*Метод Квайна визначається наступним алгоритмом*.

БФ f(x1…xn), що мінімізується, записується у вигляді ДДНФ. Починаючи з дднф (позначимо її f0) будується послідовність днф f0, f1, f2, ..., fі до тих пір, доки дві ДНФ fк та fк+1 не співпадуть між собою. При цьому перехід від форми fі до форми fі+1 (і=) відбувається згідно з наступним правилом: в формі fі виконуються всі можливі операції неповного склеювання, які застосовуються до елементарних кон’юнкцій довжини n-i, де n – кількість змінних, від яких залежить дана БФ, а і – крок, на якому виконується операція неповного склеювання; далі виконуються всі можливі операції поглинання. Результатом застосування алгоритму Квайна є скорочена ДНФ.

Легко бачити, що застосувавши алгоритм Квайна, ми отримаємо скорочену ДНФ не пізніше, ніж за n кроків.

Приклад. Продемонструємо роботу алгоритму Квайна на прикладі БФ f, що задана своєю ДДНФ: .

Операцію неповного склеювання можна застосувати до першої і другої, першої і третьої, а також до першої і четвертої констітуент 1. В результаті ми отримаємо:

Після чотирикратного застосування операції поглинання форма перетворюється в форму

Оскільки операція неповного склеювання далі застосована бути не може, то форма співпадає з формою і, в силу метода Квайна, форма - скорочена ДНФ БФ . Вона містить в собі чотири простих імпліканти. Для подальшої мінімізації БФ необхідно побудувати ІТ, знайти ядро і т.д. 

Метод Блейка. На вхід метода Блейка поступає ДНФ. Метод базується на використанні *операції узагальненого склеювання*, яка має вигляд:

АСB=ACBAB.

Справедливість цього твердження доводиться наступним чином:

АСB= ACАВСBAB = ACBAB(С) =ACBAB 

*Метод Блейка полягає в наступному*: до довільної ДНФ БФ f застосовується всі можливі операції узагальненого склеювання, а потім - всі можливі операції поглинання. В результаті ми отримуємо скорочену ДНФ даної БФ f.

Приклад. Покажемо роботу алгоритму Блейка на прикладі бф, що задана своєю днф: f(x,y,z)=xyzхz.

Можна бачити, що перший і другий члени форми f допускають узагальнені склеювання як по х, так і по y, але виникаючі в результаті склеювання елементарні кон'юнкції дорівнюють нулю, тому що вони обов'язково містять одну з букв х чи у разом з її запереченням. Нетривіальне узагальнене склеювання можливе для другого і третього членів форми. Виконуючи це склеювання, ми приходимо до ДНФ: f1= xyzхzyz.

У цій формі нетрівіальне узагальнене склеювання можливо застосовувати до першої і четвертої елементарних диз’юнкцій. Однак це склеювання приводять до елементарної кон’юнкції, що вже існує у формі f1. Таким чином, можна вважати, що у формі f1 виконані всі можливі в ній операції узагальненого склеювання. Виконавши операцію елементарного поглинання, отримаємо наступну скорочену ДНФ: f2= x хzyz. 

Приклад. Зробимо всі необхідні перетворення без будь-яких додаткових пояснень:

f=xyxzy=xyxzyxyyyz = xzyxyyyzy = xzy.

Таким чином, скорочена днф функції f має вид xz  y. 

Помітимо, що в цьому прикладі ми зробили частину елементарних поглинань ще до того, як були закінчені всі операції узагальненого склеювання. Ясно, що таким прийомом можна користуватися щоразу, коли члени, що виключаються, не можуть дати ніяких нових елементарних кон’юнкцій.

Наслідок. Якщо в скороченій ДНФ деякої БФ всі прості імпліканти не містять заперечень змінних, то ця ДНФ є МДНФ.

Дійсно, до днф, що не містить жодної букви, яка входила б у неї одночасно з запереченням і без заперечення, не може бути застосовна операція узагальненого склеювання. З методу Блейка випливає, що це можливо лише в тому випадку, коли мінімальна днф співпадає зі скороченою днф. 

Використовуючи операцію узагальненого склеювання, у ряді випадків вдається відносно просто установлювати можливість викидання зі скороченої днф деяких простих імплікант. Відповідну методику продемонструємо на прикладі.

Нехай f=xzz. Легко бачити, що третій член форми f може бути отриманий в результаті застосування операції узагальненого склеювання до перших двох членів: xz= xzz. Отримане співвідношення дає можливість спростити початкове представлення БФ f, замінивши його представленням: f = xz. 

Метод Нельсона. На вхід метода Нельсона поступає КНФ. *Метод Нельсона полягає в наступному*: якщо в довільній КНФ БФ розкрити всі дужки у відповідності з першим дистрибутивним законом, а потім застосувати всі можливі операції поглинання, то в результаті отримаємо скорочену ДНФ цієї БФ f.

Приклад. Нехай БФ задана своєю КНФ: f=(x)(z)(xy)

Розкривши дужки, отримуємо: f=xy 

1. Графічні методи мінімізації БФ: діаграми Вейча.

Діаграми Вейча. Всі розглянуті раніше методи мінімізації БФ були аналітичними. На практиці для мінімізації БФ від не великого числа змінних (не більше шести) виявилися ефективними методи, що називають *графічними*. До цих методів відносяться, зокрема, *діаграми Вейча* (карти Карно), які являють собою різновид таблиць істинності. Приклади діаграм Вейча для функцій 4-х і 5-ти аргументів представлені нижче у цьому розділі.

Кожному кортежу (двійковому набору значень змінних) відповідає одна клітка діаграми Вейча. Наборам (0,0), (0,0,0), ... і відповідним їм констітуентам одиниці 12, 123, ... зіставляються клітки з номером 0; наборам (0,1), (0,0,1),… і відповідним їм константи 1х2, 12х3, … - клітки з номером 1; набором (1,0), (0,1,0),… і відповідним їм констітуентам х12, 1х23, … - клітки з номером 2 і так далі. Якщо на даному кортежі БФ приймає значення 1, то у відповідній даному кортежу клітині записується 1. Клітки, що відповідають кортежам, на яких БФ дорівнює 0, або залишаються пустими, або заповнюється 0.

Аналогічно можуть бути побудовані діаграми і для більшого числа аргументів. Однак через те, що в цьому випадку алгоритм пошуку мінімальних днф істотно ускладнюється, діаграми для шести і більше аргументів не одержали поширення на практиці.

Із визначення операції склеювання випливає, що вона може бути застосована тільки до констітуент 1 (у загальному випадку - до елементарних кон’юнкцій), у яких усі букви, за винятком однієї, співпадають. Такі констітуенти називають *сусідніми*. Наприклад, х1х2 можна склеїти з х2 (вони є сусідніми), але не можна склеїти з х3. З розподілу констітуент на діаграмах Вейча для n=2, 3, 4 випливає, що всі сусідні констітуенти мають на діаграмах Дейча сусідні клітки. Виключення складають клітини, розташовані на границях діаграми. Для усунення цього недоліку умовно ототожнюють протилежні границі (верхню і нижню, ліву і праву) діаграми, тобто діаграму умовно “згортають” в циліндр. Будь-яким двом сусіднім кліткам відповідає елементарна кон’юнкція, що є загальною частиною відповідних двох констант і має на одну букву менше. Будь-яким чотирьом сусіднім клітинам відповідає елементарна кон’юнкція, що є загальною частиною відповідних чотирьох констітуент і має на дві букви менше і так далі. Тому склеювати можна сусідні 2, 4, 8, 16 (тобто ступені двійки) і так далі констітуент. *Пошук МДНФ за допомогою діаграм Вейча зводиться до визначення мінімального числа найбільш коротких елементарних кон’юнкцій, які покривають всі одиниці на діаграмі даної БФ*.

При n5 пошук простих імплікант на діаграмі Вейча вимагає деякого навику, оскільки в цьому випадку не всі констітyенти, що склеюються, займають сусідні клітки.

Приклад. 1. Контітуенті 1 відповідає наступна клітина, позначена 1 на діаграмі Вейча:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | w |
|  |  | 1 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

2. Діаграмі Вейча

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | w |
|  |  |  |  |  |  |  |  |
|  |  |  |  | 1 |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

відповідає констітуента 1 .

3. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  |  |  |  |  |  |  |
|  |  |  |  |  | 1 |  | w |
|  |  |  |  |  | 1 |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється .

4. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  |  |  |  | 1 |  |  |
|  |  |  |  |  |  |  | w |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  | 1 |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється

5. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  |  |  |  |  |  |  |
|  |  | 1 |  |  |  |  | w |
|  |  | 1 |  |  |  |  |  |
|  |  | 1 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється .

6. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  | 1 |  |  |  |  |  |
|  |  | 1 |  |  |  |  | w |
|  |  | 1 |  |  |  |  |  |
|  |  | 1 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється .

7. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  | 1 |  |  | 1 |  |  |
|  |  |  |  |  |  |  | w |
|  |  |  |  |  |  |  |  |
|  |  | 1 |  |  | 1 |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється .

8. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  | 1 | 1 | 1 | 1 |  |  |
|  |  |  |  |  |  |  | w |
|  |  |  |  |  |  |  |  |
|  |  | 1 | 1 | 1 | 1 |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється .

9. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  | 1 |  |  | 1 |  |  |
|  |  | 1 |  |  | 1 |  | w |
|  |  | 1 | 1 |  | 1 |  |  |
|  |  | 1 | 1 |  | 1 |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

являється .

10. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | z |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  | y |  | 1 | 1 | 1 | 1 |  |  |
| x |  |  |  |  |  |  |  | q |
|  |  |  |  |  |  |  |  |  |
|  |  |  | 1 | 1 | 1 | 1 |  |  |
|  | y |  | 1 | 1 | 1 | 1 |  |  |
|  |  |  |  |  |  |  |  | q |
|  |  |  |  |  |  |  |  |  |
|  |  |  | 1 | 1 | 1 | 1 |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | w |  |  |  |  |

являється .

11. Мінімальною д.н.ф. для булевої функції, діаграма Вейча якої має вигляд

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | y |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| x |  | 1 | \* |  |  |  |  |
|  |  | 1 | \* |  |  |  | w |
|  |  | 1 | \* |  | 1 |  |  |
|  |  | 1 | \* |  | \* |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  | z |  |  |  |  |

де \*{0, 1}, являється .

1. Проблема факторизації, факторний алгоритм.

2. Мінімізація систем БФ.

Проблема факторизації. Факторний алгоритм. Методи мінімізації БФ, що розглядалися раніше, обмежувалися вирішенням лише *канонічної* задачі мінімізації БФ, тобто задачі знаходження МДНФ заданої БФ. Однак МДНФ не у всіх випадках дає самий простий вираз для БФ в БА, тобто *МДНФ допускає подальше скорочення*. Дійсно, з наслідку методу Блейка випливає, що БФ =xyxz – МДНФ, яка представляє БФ за допомогою трьох операцій. Якщо винести х за дужки, ми знайдемо для цієї бф вираз, в якому використовуються лише дві операції: =х(yz). Остання форма уже не є ДНФ, однак її реалізація приводить до більш простої комбінаційної схеми, тому пошук подібних форм представляє собою велику практичну цінність.

Очевидно, найбільш цікава задача – це пошук *абсолютно мінімальної форми*, тобто форми, що містить в собі мінімальне число операцій заданої функціонально повної системи (див. далі). В загальному випадку, ця задача не вирішена. Вірніше, на сьогодні не існує алгоритму більш простого, ніж метод перебору всіх варіантів рішення цієї задачі. Більш того, відомо, що абсолютно мінімальну форму не завжди можна отримати із мднф, якщо використовувати лише звичайні прийоми винесення спільних множників за дужки. В той же час, пошук форми, близької до абсолютно мінімальної – реальна і практично досяжна задача. Ця задача отримала назву *проблеми факторизації*. В загальному випадку проблема факторизації також не вирішена, однак для булевого базису в ряді випадків, використовуючи операції винесення за дужки спільних множників, можна отримати форму, простішу за МДНФ даної БФ. Процес факторизації може бути описаний за допомогою спеціального алгоритму, який отримав назву *факторного алгоритму*. Його роботу розглянемо на прикладі.

Нехай мднф бф f має вигляд:

1. Позначаємо кожну елементарну кон’юнкцію заданого виразу буквами A, B, C, D і знаходимо всі попарні перетини елементарних кон’юнкцій:

, , ,

, ,

1. Вибираємо ту пару елементарних кон’юнкцій, перетин яких дає елементарну кон’юнкцію найбільшої довжини. В нашому прикладі можна вибрати або пару {A, C}, або {A, D}, або {B, D}, або {C, D}. Для визначеності вибираємо пару {C, D}.
2. Записуємо БФ аналітично, виносячи спільний множник за дужки по відношенню до пари {C, D}. В результаті отримаємо
3. Позначаємо через А1, В1, С1 диз’юнктивні члени виразу 1 і знаходимо всі їх перетини. В результаті отримаємо: , ,
4. Знову вибираємо ту пару елементів, перетин яких дає елементарну кон’юнкцію найбільшої довжини. Для визначеності вибираємо пару {A1, C1}.
5. Записуємо БФ 2 аналітично, виносячи спільний елемент за дужки по відношенню до пари {A1, C1}:

7. Нарешті, позначаємо через А2 і В2 диз’юнктивні члени БФ і знаходимо їх перетин . Остаточно маємо:

На відміну від мднф , яка містить 14 операцій типу {&, } і п’ять інверсій {¯}, отримана форма містить 9 операцій типу {&, } і три операції інверсії {¯}. 

Мінімізація систем БФ. Відзначимо, якщо провести мінімізацію БФ, що входять в деяку систему, незалежно одна від одної, то комбінаційна схема, побудована на основі цієї системи, буде складатись із ізольованих підсхем. Цю комбінаційну схему можна спростити за рахунок об’єднання ділянок підсхем, що реалізують однакові елементарні кон’юнкції із різних БФ системи.

Задача мінімізації систем БФ добре досліджена в деяких класах функціонально повних систем. Розглянемо один із найбільш розповсюджених методів мінімізації систем БФ в булевому базисі {&, , ¯}.

Нехай задана система БФ, що представлені в вигляді ДНФ:

,

,

.

Всі різні елементарні кон’юнкції системи БФ об’єднаємо в множину А, яку називають *повною множиною елементарних кон’юнкцій систем БФ*. Для нашого прикладу

.

*Сума рангів* (число букв) елементарних кон’юнкцій множини А є зручним критерієм оцінки складності заданої системи БФ.

*Система ДНФ БФ називається* *мінімальною*, якщо її повна множина елементарних кон’юнкцій містить мінімальну кількість букв, а кожна ДНФ системи включає мінімальне число елементарних кон’юнкцій найменшого рангу.

Відзначимо, ДНФ представлення БФ в мінімальній системі в загальному випадку *не співпадає* з її МДНФ.

Мінімізація систем БФ відбувається згідно з наступним алгоритмом, який представляє собою *модифікацію алгоритму Квайна*.

Нехай система із двох БФ задана своїми таблицями істинності:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | x2 | x3 | f1 | f2 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 |

Тоді:

1. Представляємо кожну із БФ системи у вигляді ДДНФ:
2. Будуємо повну множину А елементарних кон’юнкцій отриманої системи, приписуючи в дужках після кожної констітуенти 1 ознаку (у вигляді номеру чи номерів БФ у системі) її належності до тих чи інших БФ системи.
3. Будуємо ДДНФ нової функції , констітуентами 1 якої є всі елементи множини А :
4. Виконуємо мінімізацію функції , застосовуючи модифікований алгоритм Квайна. Модифікація полягає в тому, що:

а) при виконані операції неповного склеювання двох констітуент 1, елементарній кон’юнкції, що виникає в результаті склеювання, приписується ознака, яка складається із номерів БФ, спільних для двох констітуент (останнє також справедливо для двох елементарних кон’юнкцій, що склеюються). Якщо ознаки констітуент 1 не містять спільних номерів, то склеювання не відбувається;

б) операція поглинання відбувається тільки для елементарних кон’юнкцій з однаковими ознаками.

Отримані в результаті склеювання і поглинання елементарні кон’юнкції називаються *простими імплікантами системи БФ*.

Для зручності виконання операції неповного склеювання пронумеруємо кожну констітуенту 1 із ДДНФ функції :

і виконуємо склеювання.

15=

24=

34=

56=

Після проведення всіх поглинань отримуємо

Подальші склеювання і поглинання неможливі, тобто ми отримали всі прості імпліканти системи БФ і остання форма – це скорочена ДНФ БФ .

5. Будуємо ІТ з тією лише відмінністю, що для кожної констітуети 1 виділяються стільки стовпчиків, скільки різних номерів БФ містить її ознака; строчки ІТ позначаються простими імплікантами скороченої ДНФ; “\*” в ІТ проставляються згідно з наступним правилом:

а) проста імпліканта і констітуента 1 мають спільні ознаки;

б) проста імпліканта накриває констітуенту 1;

в) “\*” проставляються лише у клітини, що знаходяться на перетині строчок (з простими імплікантами) і стовпчиків (з констітуентами 1), які мають спільні ознаки.

Далі знаходимо ядро БФ і так далі.

Для нашого прикладу ІТ буде мати наступний вигляд:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | |  | |  |  |  |  |
| 1 | 2 | 1 | 2 | 1 | 1 | 2 | 2 |
|  |  | \* |  |  |  |  | \* |  |
|  |  |  | \* |  |  | \* |  |  |
|  |  |  |  |  | \* | \* |  |  |
|  |  |  |  |  |  |  | \* | \* |
|  | \* | \* |  |  |  |  |  |  |
|  |  |  | \* | \* |  |  |  |  |

Безпосередньо із ІТ випливає, що ядром функції є прості імпліканти Легко бачити, імпліканти із ядра накривають всі констітуенти 1 із ДДНФ функції . Значить, МДНФ функції  має вигляд:

Із МДНФ функції мінімальна система БФ отримується згідно з наступним *правилом*: в функцію включаються всі прості імпліканти з ознакою . Для нашого прикладу маємо:

Отримана мінімальна система БФ служить основою для побудови комбінаційної схеми. 

1. Функціонально повні системи БФ.

2. Теорема про представлення БФ через диз’юнкцію, кон’юнкцію та заперечення.

3. Теорема про функціональну повноту деяких поширених систем БФ.

4. БФ, що зберігають константу 0 (клас Р0) та 1 (клас Р1), самодвоїсті (клас S), монотонні (клас М) та лінійні (клас L) БФ.

5. Власні та замкнуті класи БФ (класи Поста).

6. Теорема про власність і замкнутість класів Р0, Р1, S, M і L.

Функціонально повні системи БФ. Система БФ називається *функціонально повною* (ФП), якщо будь-яку БФ можна представити у вигляді суперпозиції функцій цієї системи.

Теорема 9 (про представлення БФ через , &, ¯). Будь-яка БФ може бути представлена у вигляді суперпозиції {, &, ¯}, причому знак “¯” стоїть лише безпосередньо над деякими із змінних і не стоїть над внутрішніми функціями.

Доведення. Доведення будемо проводити методом математичної індукції по числу аргументів БФ.

1. Перевіримо твердження теореми при . Згідно з теоремо 2 існує різних БФ однієї змінної. Помітимо, що:

Звідси видно, що будь-яка БФ однієї змінної може бути представлена через , &, ¯, при чому знак “¯” стоїть лише над змінними. Це означає, що твердження теореми справедливо при .

1. Припустимо, що твердження теореми справедливо для всіх БФ від
2. Доведемо твердження теореми для Розглянемо довільну БФ аргументу і розкладемо її по змінній хк+1:

(\*)

Однак, якщо зафіксувати один із аргументів БФ , то ми отримаємо БФ від аргументів. Згідно з припущенням індукції будь-яка БФ від аргументів може бути представлена через , &, ¯. Значить, права частина (\*) представляє собою суперпозицію саме цих трьох БФ. Значить, твердження теореми справедливе при і, значить, згідно з методом математичної індукції, твердження теореми справедливе для будь-якого 

Теорема 10. Наступні системи БФ є функціонально повні:

1) {, &, ¯};

2) {, &, ¯};

3) {, ¯};

4) {&, ¯};

5) {, ¯};

6) {|};

7) {}.

Доведення. Функціональна повнота першої системи доведена в попередній теоремі. Цей результат можна використати для доведення ФП інших систем, наведених у теоремі.

Дійсно, якщо ми зможемо виразити через {, &, ¯}, то ми доведемо ФП системи 2. Це можна зробити на основі тотожності (29):

.

ФП інших систем доводиться на основі тотожностей:

;

;

;

;

. 

Тепер від “розрізнених і випадкових” прикладів ФП систем БФ перейдемо до їх вичерпного опису. Перш ніж одержати такий опис, розглянемо необхідні попередні поняття і вивчимо деякі їх властивості.

Говорять, що БФ f(х1,...,xn) *зберігає константу* 0, якщо f(0,...,0)=0.

Говорять, що БФ f(х1,...,xn) *зберігає константу* 1, якщо f(1,...,1)=1.

БФ f\*(x1,...,xn) називають *двоїстою* функцією для БФ f(х1,...,хn) якщо БФ f називається *самодвоїстою*, якщо f\*=f.

Введемо на множині {0,1} відношення порядку, наступним чином: 00, 01, 11. БФ називають *монотонною*, якщо для будь-яких =(1, ..., n) і =(1, ..., n), де і, і{0,1} із іі,…, nnf()f().

БФ f(х1,...,xn) називають *лінійною*, якщо її можна представити у виді наступного виразу, який називають поліномом Жигалкіна степеню не вище першого:

ai{0,1}

Позначимо класи БФ, що зберігають константу 0, константу 1, само двоїстий, монотонний і лінійний, відповідно, через Р0, Р1, S, M, L.

Класи Р0, Р1, S, M, L інколи називають *передповними* класами БФ.

Клас БФ називається *власним*, якщо він не пустий і не співпадає з класом усіх БФ.

Клас БФ називається *замкнутим* (або *класом Поста*), якщо він разом із усіма своїми функціями містить будь-яку їхню суперпозицію.

Теорема 11. Класи P0, P1, S, M, L є власними і замкнутими класами БФ.

Доведення. Перевіримо спочатку що всі ці класи є власними. Наведемо функції, які належать і не належать кожному з розглянутих класів (перевірити самостійно наведені твердження). Як класу Р0, так і класу Р1 належить, наприклад, кон’юнкція і не належать заперечення. Кон’юнкція не є самодвоїстою функцією, а заперечення є функція самодвоїста (дійсно, ). До класу монотонних належать & і не належить імплікація . До класу L належать  і не належить.

Покажемо тепер замкнутість цих класів.

Нехай f1, f2, f3P0, тобто f1(0,0)=f2(0,0)=f3(0,0)=0. Розглянемо довільну їх суперпозицію і значення суперпозиції на нульовому наборі (0,0): .

Аналогічно доводиться замкнутість класу P1.

Нехай тепер f1, f2, f3S, тобтоf1(x1,x2)=f1(x1,x2), f2(x1,x2)= f2(x1,x2), f3(x1,x2)= f3(x1,x2) і g(x1,x2)=f1(f2(x1,x2), f3(x1,x2)). Тоді

g\*(x1,x2)=g1(x1,x2)=f1(f2(x1,x2), f3(x1,x2)) = f1(f2(x1,x2),f3(x1,x2)))= =f1(f2(x1,x2), f3(x1,x2))= g(x1,x2), тобто gS.

Нехай далі f1, f2, f3М, а1b1, a2b2 і g(x1,x2)=f1(f2(x1,x2), f3(x1,x2)). Тоді g(a1,a2)= f1(f2(a1,a2), f3(a1,a2)) f1(f2(b1,b2), f3(b1,b2))=g (b1,b2). Значить, gМ.

Нехай f1, f2, f3L і, наприклад, f1(x1,x2)=а0а1х1а2х2, f2(x1,x2)=b0b1х1b2х2, f3(x1,x2)=c0c1х1c2х2. Тоді

g(x1,x2)= f1(f2(x1,x2), f3(x1,x2))=a0a1f2(x1,x2)a2f3(x1,x2)=

=a0a1(b0b1х1b2х2)a2(c0c1х1c2х2)=

=(а0а1х1а2х2)(a1b1a2c2)x1(a1b2a2c2)x2= d0d1х1d2х2, тобто gL



Відзначимо, що існують функції, які належать кожному із п’яти класів, наприклад, f(x)=x. 

1. Теорема Поста про функціональну повноту.

2. Наслідки теореми Поста про функціональну повноту.

Теорема Поста (про функціональну повноту). Система БФ F={f1, f2, …, fk,…} є функціонально повною тоді і тільки тоді, коли в цій системі існує:

1) хоча б одна функція, що не зберігає константу 0;

2) хоча б одна функція, що не зберігає константу 1;

3) хоча б одна не самодвоїста;

4) хоча б одна не лінійна;

5) хоча б одна немонотонна функція.

Доведення. Нехай система F є ФП. Припустимо, що всі БФ цієї системи входять у клас Р0. Оскільки, згідно з попередньою теоремою, клас Р0 є власним і замкнутим, то знайдеться хоча б одна БФ, що не належить цьому класу. Значить, система F не є ФП, що входить в протиріччя з початковим припущенням. Значить, в системі F повинна бути хоча б одна функція, яка не належить класу Р0.

Аналогічно доводиться існування хоча б по одній функції, що не належать класам Р1, S, M, L.

Достатність. Припустимо, що серед функцій системи F є такі БФ f0, f1, f2, f3, f4, що f0Р0, f1Р1, f2S, f3M, f4L. Покажемо, що за допомогою цих функцій і змінних х1, ..., хn можна побудувати (за допомогою операції суперпозиції) будь-яку БФ f(х1, ..., хn) цих змінних. З цією метою зведемо систему функцій f0, f1, f2, f3, f4, що задовольняють умовам теореми, до деякої функціонально повної системи, наприклад, {, ¯} або {&, ¯}.

Побудуємо спочатку з функцій, що входять в умови 1, 2, 3 теореми й змінних x1,…,xn константи 0 і 1.

Розглянемо функцію f0(х1, ..., хn)і введемо функцію g(x1)=f0(x1, …, x1) як суперпозицію f0 і х1.

За умовою g(0)=1. Далі можливі два варіанти.

а) g(1)=1. Тоді g(x1)1 і, маючи функцію f1, можна отримати константу 0.

б) g(1)=0. Тоді g(x1)=x1.

У випадку б) розглянемо БФ f2S. Оскільки f2 не самодвоїста, то існують два протилежних набори значень її аргументів =(1,…,k) і =(1,…,k) такі, що f2()=f2(). Побудуємо суперпозицію h(x1) функцій f2 і і(x1) таку, що: h(x1)= f2(1(x1),…, к(x1)), де і(x1)=х1, якщо і=0 і і(x1)=х1, якщо і=1. Тоді:

h(0)=f2(1(0), …, к(0))=f2(1,…,k)=f2(1,…,k)=f2(1(1), …, к(1))=h(1),

тобто h(0)=h(1) і, отже, h(x1)0 або h(x1)1. Утворюючи суперпозицію цієї константи з раніше побудованою функцією g(x1)=x1 ми отримаємо другу константу.

Далі, вибравши немонотонну функцію f3(x1,…,xr)М, побудуємо заперечення. У силу не монотонності функції f4 знайдеться пара наборів =(1,…,r) і =(1,…, r) таких, що , але f3()>f3(). Перевіряючи значення функції f3 на проміжних наборах  (), можна виділити пару сусідніх наборів к іk+1 , що відрізняються значенням i-їкомпоненти:

k=(1, …, i-1, 0, i+1, …, r), **k+1=(1, …, i-1, 1, i+1, …, r), k<k+1 і f3(k)>f3(к+1). Підставляючи у функцію f3(x1,…,xr) замість змінної хj константу j (j=1, 2, …, i-1, i+1, …, r) ми отримаємо функцію однієї змінної: q(xi)=f3((1, …, i-1, xi, i+1, …, r). У силу прийнятих угод: q(0)=1 і q(1)=0. Отже q(xi)=xi, і=1, 2, ..., n.

Для завершення доказу теореми нам залишилося одержати функцію х1&х2 або x1x2.

Розглянемо функцію f4(х1,…,хр). Побудуємо для цієї функції канонічний поліном Жигалкіна. У цьому поліномі обов'язково існує компонента, що містить добуток яких-небудь двох змінних хi і xj. Не порушуючи спільності, можна вважати, що xi=x1, xj=x2.

Всі члени поліному функції f4 можна згрупувати в такий спосіб:

f4(х1,…,хр)=х1х2Ф1(х3,…,хр)х1Ф2(х3,…,хр)х2Ф3(х3,…,хр)Ф4(х3,…,хр).

Функція Ф1  0. Вибираючи значення х3,…,хр так, щоб Ф1 прийняла значення 1, ми перейдемо до функції: (х1, х2)=х1х2х1х2, де , ,  - константи 0 або 1.

Розглянемо суперпозицію:

(х1+b,х2+а)=(х1)(х2)(х1)(х2)*=*х1х2, де =.

Побудована функція  дорівнює х1х2, якщо = 0 і дорівнює =х1x2, якщо =1.

Тобто, систему БФ f0Р0, f1Р1, f2S, f3M, f4L, що входять в умови 1-5 теореми, вдається звести до функціонально повної системи {, ¯} або {&,¯}. 

Наслідок 1. У результаті суперпозиції довільної немонотонної функції f(x1,…,xn) с константами 0 і 1 може бути отримана функція запереченняxi одного з аргументів функції f(x1,…,xn).

Доказ. Випливає з доведення теореми Поста.

Система БФ називається *ослаблено функціонально повною*, якщо ця система містить в собі константи 0 і 1.

Наслідок 2 (теорема Поста про ослаблену функціональну повноту). Для того, що б система БФ була ослаблена функціонально повною необхідно й достатньо, щоб ця система містила хоча б одну нелінійну і хоча б одну немонотонну функцію.

Доказ. У випадку наявності констант 0 і 1 у нашому розпорядженні виявляється функція, що не зберігає 0 (функція 1), функція, що не зберігає 1 (функція 0), і несамодвоїста функція (кожна з функцій 0 і 1). Разом з тим константи є, очевидно, лінійними й монотонними. 

Визначення. Будемо називати функціонально повну систему БФ *нескоротною*, якщо будь-яка її власна частина функціонально повною уже не є.

Будь-яку функціонально повну систему БФ можна привести до нескоротного виду, викидаючи з неї зайві функції. Як випливає з теореми Поста, в будь-якій нескоротній функціонально повній системі БФ існує не більше п’яти функцій. Насправді їхнє число може бути скорочене до чотирьох, оскільки функція f, що не зберігає 0, або не зберігає 1, або (якщо f(1,...,1)=1) є несамодвоїстою. Таким чином, крім цієї функції, досить залишити в системі три функції: лінійну, немонотонну і функцію, що не зберігає 1 (якщо f(1,...,1)=1), або несамодвоїсту функцію (якщо f(1,...,1)=0).

Разом з тим неважко привести приклад функціонально повної нескоротної системи, що складається із чотирьох функцій. Такою системою є, наприклад, система функцій: f1=0, f2=1, f3=x1x2, f3=x1x2x3.

Із цієї системи не можна виключити жодної функції, оскільки функція f1 є єдиною функцією цієї системи, що не зберігає константу 1, f2 - єдиною функцією, що не зберігає константу 0, f3 - єдиною нелінійною функцією, а f4 - єдиною немонотонною функцією (несамодвоїстими тут є три функції f1, f2 і f3). Таким чином, ми приходимо до наступного

Наслідок 3. Максимально можливе число БФ у нескоротній функціонально повній системі БФ дорівнює чотирьом.

Наслідок 4. На практиці, для знаходження ФП систем бф, зручно користуватися таблицями, подібними наведеній нижче. Очевидно, критерієм функціональної повноти у цьому випадку буде наявність «+» у п’ятьох стовпчиках «Групи елементів» таблиці.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Тип елементу | Функція | Групи елементів | | | | |
| Немоно-тонні | нелінійні | несамо-  двоїсті | не зберіга-  ють 0 | не зберіга-  ють 1 |
| Тривіальний | х | - | - | - | - | - |
| Нулевий | 0 | - | - | - | - | + |
| Одиничний | 1 | - | - | - | + | - |
| Інвертор | х | + | - | - | + | + |
| Кон’юнкція | хy | - | + | + | - | - |
| Диз’юнкція | хy | - | + | + | - | - |
| Співпадання  з забороною | хy    | + | + | + | - | + |
| Розділення з  забороною  (імплікація) | хy   xy | + | + | + | + | - |
| Співпадання з подвійною забороною, стрілка Пірса або функція Вебба | хy   xy | + | + | + | + | + |
| Розділення з подвійною забороною, (штрих Шеффера) | хy    xy | + | + | + | + | + |
| Елемент  рівнознач-ності | хуху  ху | + | - | + | + | - |
| Елемент  нерівнознач-ності | ху | + | - | + | - | + |

**3. Приклад виконання лабораторної роботи №1**

(Увага! Цей приклад показує, насамперед, схему виконання роботи - у ньому можливі технічні помилки!)

Завдання №1. Визначити свій варіант бульової функції. Для цього необхідно номер варіанту перевести в двійкову систему числення і записати сім його молодших розрядів у вигляді слова *a*7*a*6*a*5*a*4*a*3*a*2*a*1*.* Визначивши значення *ai* підставити їх у табл. 1.1.

Приклад. Нехай номер мого варіанту 12, якщо перевести у двійкову систему вийде 0001100. Тоді, згідно з умовою, *a*1=0, *a*2=0, *a*3=1, *a*4=1, *a*5=0, *a*6=0, *a*7=0. Далі будемо використовувати ці знайдені данні для заповнення таблиці 1.1.

Таблиця 1.1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Інформація | | | | | | |
| на вході | | | | | на виході | |
| X4 | X3 | X2 | X1 | f | |  |
| 0 | 0 | 0 | 0 | 0 | | 1 |
| 0 | 0 | 0 | 1 | 0 | | 1 |
| 0 | 0 | 1 | 0 | 0 | | 1 |
| 0 | 0 | 1 | 1 | 1 | | 0 |
| 0 | 1 | 0 | 0 | 1 | | 0 |
| 0 | 1 | 0 | 1 | 1 | | 0 |
| 0 | 1 | 1 | 0 | 0 | | 1 |
| 0 | 1 | 1 | 1 | 0 | | 1 |
| 1 | 0 | 0 | 0 | 1 | | 0 |
| 1 | 0 | 0 | 1 | 0 | | 1 |
| 1 | 0 | 1 | 0 | 1 | | 0 |
| 1 | 0 | 1 | 1 | 0 | | 1 |
| 1 | 1 | 0 | 0 | 0 | | 1 |
| 1 | 1 | 0 | 1 | 1 | | 0 |
| 1 | 1 | 1 | 0 | 0 | | 1 |
| 1 | 1 | 1 | 1 | 0 | | 1 |

Завдання №2. Для заданої функції і для її заперечення знайти МДНФ. Представити функцію у всіх восьми нормальних формах.

Побудуємо ДДНФ для заданої функції та її заперечення.

Мінімізуємо функцію та її заперечення методом Карно-Вейча.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X3 |  | | | |
| X4 |  | 1 |  | 1 |  |
|  |  |  | 1 | X2 |
|  |  |  | 1 |  |  |
| 1 | 1 |  |  |  |
|  | X1 |  | | |

Для :

Для :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X3 |  | | | |
| X4 | 1 |  | 1 |  |  |
| 1 | 1 | 1 |  | X2 |
|  | 1 | 1 |  | 1 |  |
|  |  | 1 | 1 |  |
|  | X1 |  | | |

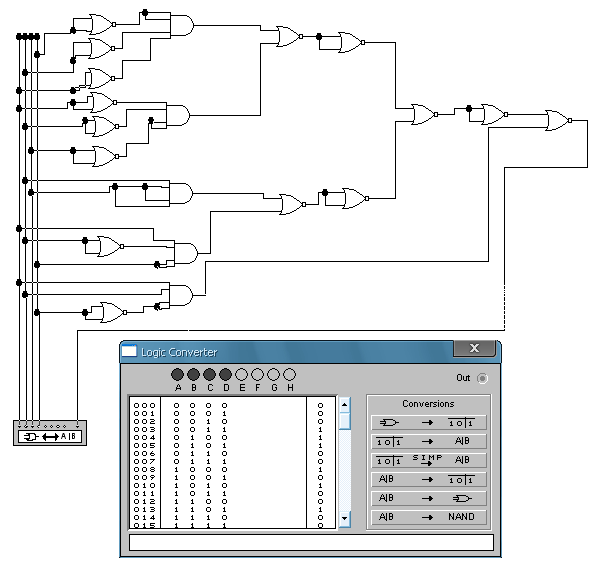
В результаті мінімізації отримуємо МДНФ функції та її заперечення. Представляємо їх у восьми нормальних формах.

Завдання №3. Одержати операторні представлення функції, що можуть бути реалізовані на елементах, заданих у табл. 1.2.

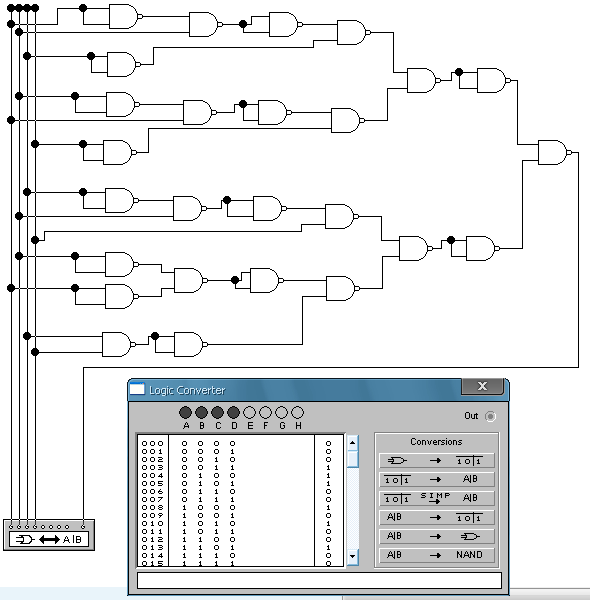
Згідно мого варіанту (12), елементи на яких я маю будувати схеми, що реалізують операторні представлення, такі: 2І-НІ, 2АБО-НІ/4І (дані з таблиці 1.2). Затримка елементів 2І-НІ та 2АБО-НІ – 22 мсек. Затримка елементів 4І – 24мсек.

Реалізуємо їх на схемах, використовуючи відповідні елементи:

На елементах 2АБО-НІ/4I:



На елементах 2І-НІ:



Завдання №4. Визначити операторну форму (із отриманих), що забезпечує одержання комбінаційної схеми з максимальною швидкодією і мінімальними затратами обладнання, тобто схему з кращим параметром T і схему з мінімальним значенням N. Усі мікросхеми в табл. 1.2 мають по 14 виводів.

Складність першої схеми та її затримка представлена нижче:

Складність другої схеми та її затримка представлена нижче:

Звідси видно, що перша схема краща за затратою часу та однакові за затратою обладнання.

Можна зробити висновок, що краще використовувати першу схему, вона вигідніша.

1. Синтез перетворювачів кодів з використанням логічних елементів.

2. Приклад.

Проектування КС з багатьма виходами відрізняється тим, що система булевих функцій піддається спільній мінімізації, а потім перетворюється до операторного представлення таким чином, щоб число використовуваних логічних елементів було мінімальним. Використовуючи дані, отримані на першому кроці, заповнюємо таблицю 1.2:

Таблиця 1.2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Інформація | | | | | | | |
| на вході | | | | на виході | | | |
| X4 | X3 | X2 | X1 | f4 | f3 | f2 | f1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |

Випишемо всі різні констітуенти одиниці заданої системи булевих функцій з ознаками належності до тієї чи іншої функції.

Мінімізація систем БФ відбувається згідно з наступним алгоритмом, який представляє собою *модифікацію алгоритму Квайна*.

Виконуємо мінімізацію функції, застосовуючи модифікований алгоритм Квайна.

Модифікація полягає в тому, що:

1) при виконані операції неповного склеювання двох констітуент 1, елементарній кон’юнкції, що виникає в результаті склеювання, приписується ознака, яка складається із номерів БФ, спільних для двох констітуент (останнє також справедливо для двох елементарних кон’юнкцій, що склеюються). Якщо ознаки констітуент 1 не містять спільних номерів, то склеювання не відбувається;

2) операція поглинання відбувається тільки для елементарних кон’юнкцій з однаковими ознаками.

Отримані в результаті склеювання і поглинання елементарні кон’юнкції називаються простими імплікантами системи БФ.

Проводимо операцію неповного склеювання серед імплікант множини А:

В – множина, яка містить в собі трійкові імпліканти, отримані в результаті неповного склеювання елементів множини А.

Проводимо операцію неповного склеювання серед трійкових імплікант множини В:

Подальші склеювання і поглинання неможливі.

В результаті виконання всіх операцій поглинання і склеювання отримуємо скорочену ДНФ системи бульових функцій:

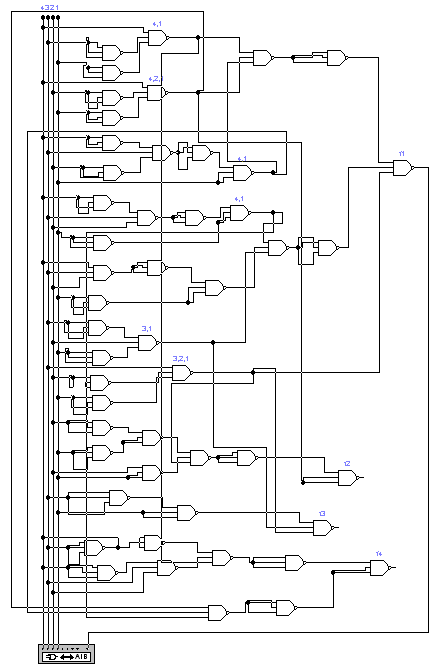
Будуємо імлікантну таблицю (див. нижче).

В цій таблиці заперечення позначенні знаком <’>.Стовпчики – це імпліканти, а рядки - конституенти 1.

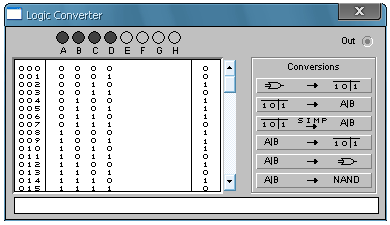
Мднф функції  і відповідна їй мінімальна система булевих функцій будуть мати вигляд.

Будуємо (самостійно) операторні представлення на елементах 3І-НІ функцій f1- f4 і по ним – схему перетворювача кодів (див. нижче).

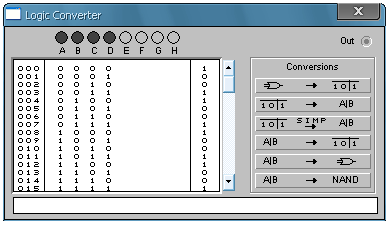
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | x2’x1’(2) | x3’x1(3) | x3’x2(3) | x2x1’(1) | x2x1(2) | x3x1’(1) | x4x3’(4) | x4x1’(1) | x3’x2x1’(3,1) | x3’x2x1(3,2) | x4’x3x2’(1) | x4’x3x1’(1) | x3x2’x1’(3,2,1) | x4’x3x1(4) | x4’x3x2(4) | x4x3’x2’(4,1) | x4x3’x1’(4,1) | x4x2’x1’(4,2,1) | x4x3’x1(4,3) | x4x3’x2(4,3) | x4’x3x2’x1(4,1) | x4’x3x2x1’(4,1) | x4’x3x2x1(4,2) | x4x3’x2’x1(4,3,1) | x4x3’x2x1’(4,3,1) | x4x3’x2x1(4,3,2) | x4x3x2’x1’(4,3,2,1) | x4x3x2x1’(1) | |
|  | 2 | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3 |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3 |  |  | \* |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  | \* |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3 |  | \* | \* |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  | \* |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 3 |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | \* |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  | \* |  |  |  |  | \* | \* | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |
|  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  | \* |  |  |  |  |  |  |
| 1 |  |  |  | \* |  | \* |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |
|  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  | \* | \* |  |  |  |  |  |  |  | \* |  |  |  |  |  |
| 2 |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |
|  | 4 |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  | \* | \* | \* |  |  |  |  |  |  |  |  |  |  |
| 2 | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  | \* | \* | \* |  |  |  |  |  |  |  |  |  |  |
|  | 4 |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  | \* |  |  | \* |  |  |  |  | \* |  |  |  |  |
| 3 |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  | \* |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  | \* |  |  |  |  |
|  | 4 |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  | \* |  |  | \* |  |  |  |  | \* |  |  |  |
| 3 |  |  | \* |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  | \* |  |  |  |
| 1 |  |  |  | \* |  |  |  | \* | \* |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  | \* |  |  |  |
|  | 4 |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  | \* | \* |  |  |  |  |  | \* |  |  |
| 3 |  | \* | \* |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  | \* | \* |  |  |  |  |  | \* |  |  |
| 2 |  |  |  |  | \* |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |
|  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  | \* |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |  |
| 2 | \* |  |  |  |  |  |  |  |  |  |  |  | \* |  |  |  |  | \* |  |  |  |  |  |  |  |  | \* |  |
| 1 |  |  |  |  |  | \* |  | \* |  |  |  |  | \* |  |  |  |  | \* |  |  |  |  |  |  |  |  | \* |  |
|  | 1 |  |  |  | \* |  | \* |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | \* |
|  | 2 |  |  |  |  | \* |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |



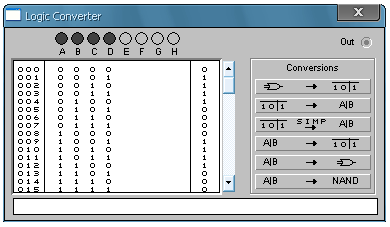
1) f1



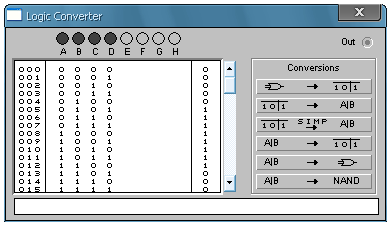
2) f2



3) f3



4) f4



Отримана комбінаційна схема реалізує всі задані функції правильно.

Складність даної схеми та її затримка представлена нижче:

1. Мультиплексори.

2. Синтез КС з використанням мультиплексорів.

Завдання №6. Побудувати схему для реалізації функції, заданої табл. 1.1, якщо крім елементів з табл. 1.2 можна використовувати мультиплексори з двома керуючими входами. Два мультиплексора розміщаються в одному корпусі мікросхеми на 16 виводів. Порівняти по складності отриману схему зі схемою, побудованою без використання мультиплексорів.

Згідно з методичними вказівками будуємо шість діаграм Вейча для визначення залишкових функцій.

.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | X3 | |  | |  |  | |  | | |
|  | |  |  | |  | |  | | |  | |  |  |
| X4 | |  |  | | 1 | |  | | | 1 | |  |  |
|  |  | |  | |  | | | 1 | |  | X2 |
|  | |  |  | |  | | 1 | | |  | |  |
|  | |  | 1 | | 1 | |  | | |  | |  |  |
|  | |  |  | |  | |  | | |  | |  |  |
|  |  | | |  | | X1 | |  |  | |  | | |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | X3 |  | |  |  |  | | |
|  | |  |  | |  |  | |  | |  |  |
| X4 | |  |  | | 1 |  | | 1Y3 | |  |  |
|  |  | |  |  | |  | |  |  |
|  |  | | Y2 |  | | 1 | |  | X2 |
|  | |  |  | |  |  | |  | |  |
|  | |  |  | | Y1 | 1 | |  | |  |
|  | |  |  | |  |  | |  | |  |  |
|  | |  | Y0 1 | | 1 |  | |  | |  |  |
|  |  | | |  | X1 | |  |  |  | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | X3 | |  |  | | |  | |  |  | | |
|  | |  |  | |  | | |  |  | |  | | |  |  |
| X4 | |  |  | | 1 | | |  | 1 | | Y2 | | |  |  |
|  |  | | Y3 | | |  |  | | 1 | | |  | X2 |
|  | |  |  | |  | | |  |  | |  | | |  |
|  | |  |  | |  | | |  | 1 | |  | | |  |
|  | |  | Y1 1 | | 1 | | |  |  | | Y0 | | |  |  |
|  | |  |  | |  | | |  |  | |  | | |  |  |
|  |  | | |  | | X1 |  | | |  | |  | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | X3 |  | | |  | |  |  | | |  | | | |
|  | |  |  | | |  |  | |  | | |  |  | | |  |  |
| X4 | |  | Y2 | | |  | 1 | |  | | |  | 1 Y2 | | |  |  |
|  |  | | |  |  | | Y3 | | |  | 1 | | |  | X2 |
|  | |  |  | | |  |  | |  | | |  |  | | |  |
|  | |  |  | | |  |  | | 1 | | |  |  | | |  |
|  | |  | Y0 1 | | |  | Y1 1 | |  | | |  | Y0 | | |  |  |
|  | |  |  | | |  |  | |  | | |  |  | | |  |  |
|  |  | | |  |  | | | X1 | |  |  | | |  |  | | |

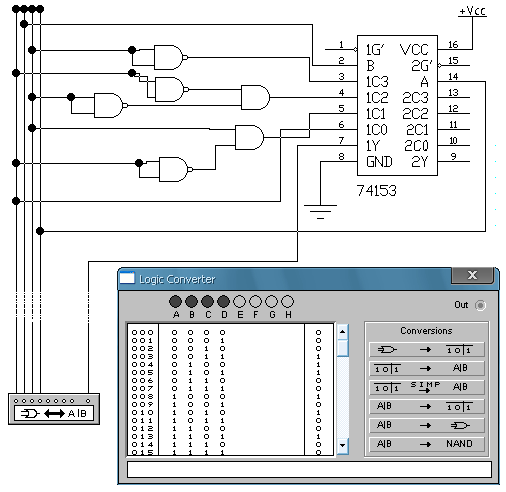
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | X3 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| X4 |  | Y0 |  | Y1 1 |  |  | 1 Y0 |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | 1 |  | X2 |
|  |  | Y2 |  | Y3 | 1 |  | Y2 |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  | Y0 1 |  | 1 | Y1 |  | Y0 |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | X1 |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | X3 | |  |  | | |  | |  |  | | |
|  | |  |  | |  | | |  |  | |  | | |  |  |
| X4 | |  |  | | Y2 1 | | |  |  | | Y0 1 | | |  |  |
|  |  | |  | | |  |  | |  | | |  |  |
|  |  | |  | | |  |  | | 1 | | |  | X2 |
|  | |  |  | | Y3 | | |  | 1 | | Y1 | | |  |
|  | |  |  | |  | | |  |  | |  | | |  |  |
|  | |  | Y2 1 | | 1 | | |  |  | | Y0 | | |  |  |
|  | |  |  | |  | | |  |  | |  | | |  |  |
|  |  | | |  | | X1 |  | | |  | |  | | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | X3 |  | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  |
| X4 |  |  |  | Y3 1 |  |  |  | Y0 1 |  |  |
|  |  |  |  |  | Y1 |  | 1 |  | X2 |
|  |  |  |  |  |  | 1 |  |  |  |
|  |  | Y2 1 |  | 1 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  | X1 |  |  |  |  | | |

Вибираємо розклад по змінним Х3, Х1, так як цей розклад містить у собі найменшу кількість змінних.

Тепер на елементах заданих у таблиці 1.2 та за допомогою мультиплексора ми будуємо схему цієї булевої функції.



Дана комбінаційна схема реалізує задану функцію правильно.

1. Дешифратори.

2. Синтез КС з використанням дешифраторів.

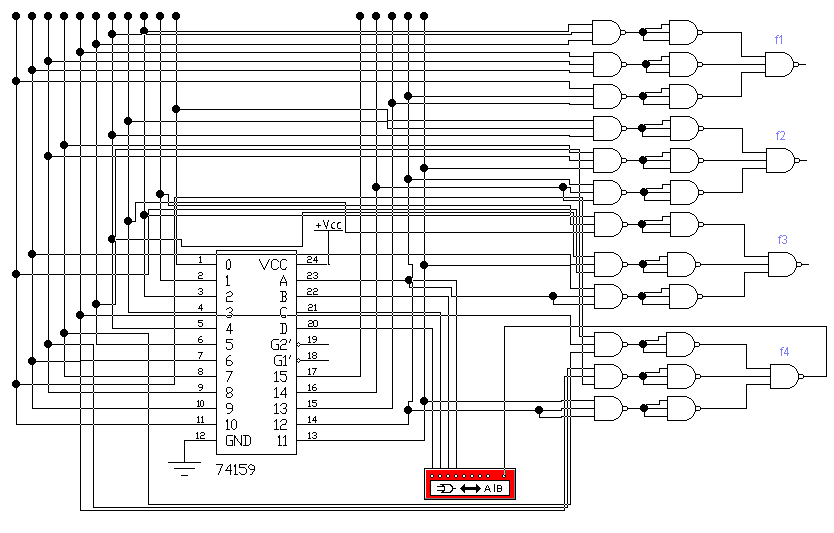
Завдання №7. Побудувати перетворювач кодів по табл. 1.3 з використанням елементів ЗІ-НІ і чотирьохвходового дешифратора з інверсними виходами. Дешифратор розміщений у корпусі з 24 виводами. Порівняти значення *N* для схеми з дешифратором і без дешифратора.



Рис. 1.2

Виписуємо з таблиці 1.3 констітуенти одиниці. Застосовуємо закон Де – Моргана до всіх чотирьох функцій:

Будуємо схему, реалізуючу ці чотири функції на елементах 3І-НІ та з використанням дешифратора.



Отримана комбінаційна схема реалізує всі задані функції правильно.