Лабораторная работа №3:

***«Построение логических схем»***

***Цель работы:*** освоить алгоритм построения таблиц истинности для логических функций; научиться определять и анализировать функции проводимости переключательных схем.

***Общие сведения из теории***

Любая переключательная функция может быть выражена через функции

―И", "ИЛИ", "НЕ". Схема, выражающая функцию F через функции ―И", "ИЛИ", "НЕ", называется ее **логической схемой**.

# Этапы построения логической схемы:

1. составляется таблица истинности;
2. по таблице истинности строится логическая функция с помощью СДНФ (совершенной дизъюнктивной нормальной формы);
3. по возможности полученная формула минимизируется;
4. если заданы базисные элементы, то с помощью законов Моргана приводится к заданному базису.

Логическая схема функции F **= (А + С) (А + D) (В + С) (В + D)** представится следующим образом:

A B C D

или

или

A+D

и

или

или

A+C

F

B+C

B+D

Однако если предварительно упростить функцию F, дважды используя

свойство распределительности, то для реализации ее потребуется вместо четырех схем "ИЛИ" и одной схемы ‖И" только две схемы "И" и одна схема "ИЛИ":

F = (А + С) (А + D) (В + С) (В + D) = (A + СD) (В + СD) = АВ + СD.

A B C D

F

и

и

или

Существует множество способов упрощения переключательных функций и определения простоты функции. Мы примем, что простейшей будет функция, содержащая наименьшее количество элементов или их отрицаний.

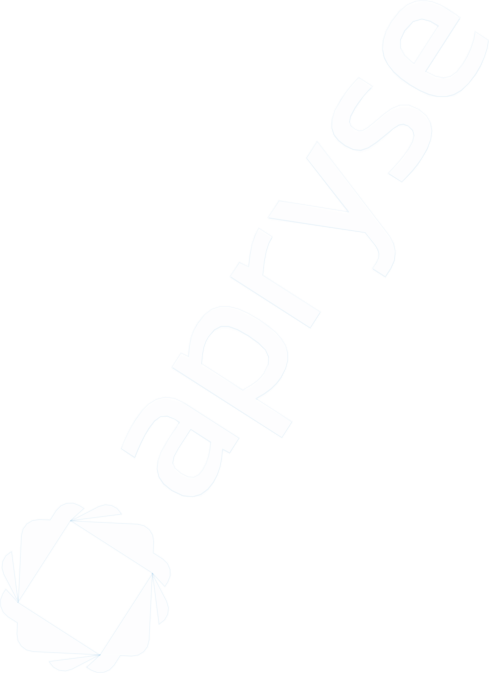
# Упрощение переключательной функции

**путем приведения ее к дизъюнктивной нормальной форме**

В основе представления ПФ в дизъюнктивных формах лежит понятие элементарной конъюнкции.

Конъюнкция любого числа двоичных переменных х1,x2,х3,...xn называется элементарной, если сомножителями в ней являются либо одиночные аргументы,

либо отрицания одиночных аргументов. Например, конъюнкции являются элементарными.



*x*1 *x*2 *x*3 ,

*x*1 *x*2 *x*4

**Дизъюнктивной нормальной формой (ДНФ)** переключательной функции называется дизъюнкция (логическая сумма) любого числа элементарных конъюнкций. Например, ПФ

записана в ДНФ, так как она представляет собой логическую сумму элементарных конъюнкций.

Число переменных, входящих в элементарную конъюнкцию, определяет

ранг этой конъюнкции. Например,

* конъюнкции 2-го ранга и т. д.

*x*1 ,

*x*2 ,

*x*3 — конъюнкции 1-го ранга;

*x*2 *x*3 ,

*x*1 *x*3

**Совершенной ДНФ (СДНФ) ПФ**, имеющей n аргументов, называется такая форма, в которой все конъюнкции имеют ранг n.

СДНФ переключательной функции записывается по таблице истинности.

# Алгоритм построения таблиц истинности для сложных выражений:

1. *Определить количество строк*:

**количество строк = 2n + строка для заголовка**, **n** - количество простых высказываний.

1. *Определить количество столбцов*:

# количество столбцов = количество переменных + количество логических операций;

* + определить количество переменных (простых выражений);
  + определить количество логических операций и последовательность их выполнения.

Порядок выполнения логических операций в сложном логическом выражении:

инверсия; конъюнкция; • дизъюнкция; +

1. *Заполнить столбцы результатами выполнения логических операций в обозначенной последовательности с учетом таблиц истинности основных логических операций*.

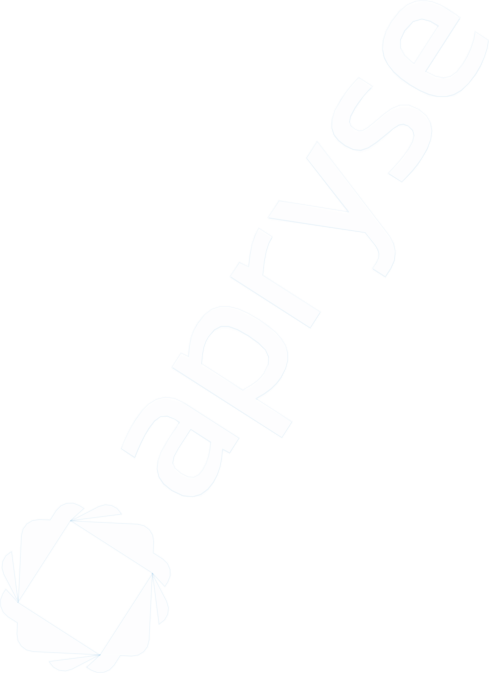
**Пример:** Составить таблицу истинности логического выражения: F=

*A*  (*B*  *C*)

1. Определить количество строк:

на входе три простых высказывания: А, В, С поэтому n=3 и количество строк

= 23 +1 = 9.

1. Определить количество столбцов:

переменные: А, В, С; логические операции:

*A* - инверсия (обозначим через 1);

B+C - операция дизъюнкции (обозначим через 2);

а также искомое окончательное значение арифметического выражения:

F= *A*  (*B*  *C*) т.е. F = 1 & 2 - это операция конъюнкции.

1. Заполнить столбцы с учетом таблиц истинности логических операций.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **1** | **2** | **F** |
| 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 |

# Алгоритм построения логической функции по ее таблице истинности:

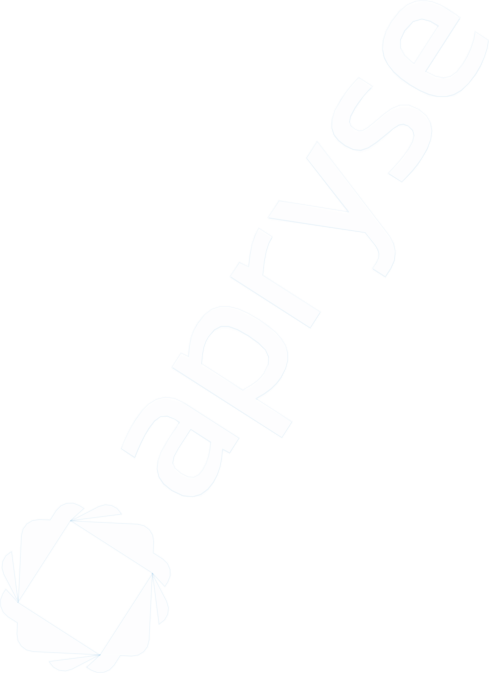
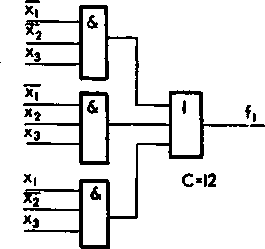
1. Выделить в таблице истинности те строки, в которых значение функции равно 1
2. Выписать искомую формулу в виде дизъюнкции нескольких логических элементов. Число этих элементов равно числу выделенных строк.
3. Каждый логический элемент в этой дизъюнкции записать в виде конъюнкции аргументов функции.
4. Если значение какого-либо аргумента функции в соответствующей строке таблице равно 0, то этот аргумент взять с отрицанием.
5. Используя правило склеивания, можно упростить ПФ, заданную в СДНФ. Для этого в СДНФ сначала склеиваются между собой конъюнкции ранга

n, затем полученные конъюнкции ранга (n — 1), (n —2), и так до тех пор, пока в выражении для ПФ не останется ни одной пары склеиваемых между собой конъюнкций. Операция склеивания позволяет понизить ранг конъюнкций и сократить их число.

**Пример:** построить логическую функцию по ее таблице истинности

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **C** | **F** |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

1. В 2, 4 и 6 строках таблицы истинности значение функции равно 1.



1. Так как строки три, получаем дизъюнкцию трех элементов: ( ) + ( ) + ( ).
2. Каждый логический элемент в этой дизъюнкции запишем в виде конъюнкции аргументов функции A, B и C:
3. Берем аргумент с отрицанием если его значение в соответствующей строке таблицы равно 0 и получаем искомую функцию:

F= ( *A*  *B*  *C* ) + ( *A*  *B*  *C* ) + ( *A*  *B*  *C* )

1. Выполним склеивание конъюнкций

*A*  *B*  *C* и

*A*  *B*  *C*

по переменной B

и конъюнкций

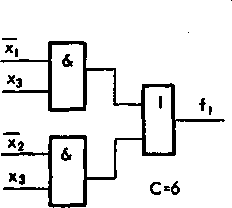
*A*  *B*  *C* и

*A*  *B*  *C*

по переменной A. В результате функция F

преобразуется к виду:

*A*  *C*  *B*  *C*



Для каждой функциональной схемы можно сделать оценку ее сложности, которая выражается ценой схемы С. Цена С определяется суммарным числом

входов логических элементов. Чем меньше величина С, тем проще функциональная схема.

# Задания к лабораторной работе

В соответствии с выданным вариантом задания преподавателем выполнить:

1. составить таблицу истинности логического выражения
2. построить логическую функцию по таблице истинности
3. по таблице истинности построить логическую функцию с помощью СДНФ и минимизировать
4. сделать оценку сложности функциональной схемы

# Контрольные вопросы:

1. Для чего используется алгебра логики?
2. Дайте определение - Двоичные переменные, Переключательные функции, Логический элемент компьютера
3. Базовые и составные логические функции- краткая характеристика
4. Назовите четыре основных закона булевой алгебры
5. Этапы построения логической схемы
6. Чем вызвана необходимость упрощения переключательной функции путем приведения ее к дизъюнктивной нормальной форме?
7. Что такое цена схемы? Как ее определить?