Лабораторная работа №2 (2019).

«Функции алгебры логики»

Цель работы: освоение методики аналитического и экспериментального исследования конечного автомата без памяти с двоичными входами и выходами, его формального описания и исследования с помощью функций алгебры логики (ФАЛ); овладение навыками аналитического оперирования с ФАЛ, реализация ФАЛ в форме комбинационной схемы.

Задание

1. *Выполнить цикл индивидуальных упражнений* по аналитическому преобразованию ФАЛ и доказательству свойств элементарных ФАЛ. Упражнения получить на сайте ws-dss.com, аналогично тому, как они были получены в первой лабораторной работе (метод **test_bool**).

2. Для заданного на ЭВМ (nх1) - полюсника с двоичными входами и выходом составить таблицу соответствия входных и выходных слов, являющуюся также таблицей истинности для заданной ФАЛ. Для получения исходных данных запустить на сайте ws-dss.com метод ***blackbox****.*

3. Исследовать аргументы полученной в п. 2 ФАЛ на существенность (фиктивность). При обнаружении фиктивных аргументов записать таблицу истинности этой ФАЛ как функции только для существенных аргументов.

4. Образовать СДНФ и СКНФ, полученной в п. 3 ФАЛ. Упростить полученную ФАЛ с использованием операций поглощения и склеивания.

5. Составить комбинационную схему из логических элементов НЕ, ИЛИ, И, соответствующую окончательной ДНФ, полученной в п. 4, оценить количественно её сложность.

6. Составить отчет, отражающий выполнение п.п. 1 - 5. Устно ответить на контрольные вопросы. Отчет по работе должен содержать задание, материалы по выполнению всех пунктов задания. В конце отчета должны быть сделаны выводы. Все трассировки с сайта ws-dss.com должны быть распечатаны полностью с QR кодами.

Методические указания

Каждому входу заданного на ЭВМ конечного автомата ставится в соответствие двоичная логическая переменная x_i (i = 1, 2, ... , n), а выходу – двоичная логическая переменная y. После этого составляется таблица, в которой перечисляются в порядке возрастания все n - разрядные двоичные входные слова – j-е входные наборы (для n = 4 см. табл. 1.). Здесь j есть десятичный номер соответствующего двоичного входного набора при чтении последнего сверху вниз.

Таблица 1.

j	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
x_1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x_2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
x_3	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
x_4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
y																

Эксперимент состоит в задании на входах "черного ящика" значений каждого входного слова < $\alpha_{1j}, \alpha_{2j}, \ldots, \alpha_{nj}$> с номером j (j = 0,1,2, ..., 2^n–1), фиксации на выходе соответствующих значений выходной переменной и заполнении нижней части табл. 1. Выходное слово **y** есть изображающее число ***#y*** соответствующей ФАЛ. Полученная таблица может рассматриваться как таблица истинности заданной n-местной ФАЛ.

Факт существенности или фиктивности одного любого аргумента $\boldsymbol{x_i}$ ФАЛ устанавливается на основе анализа множества T_0 и T_1 этой функции после исключения из наборов i -го символа. Если после его исключения в множествах T_0 и T_1 обнаружатся одинаковые, то переменная $\boldsymbol{x_i}$ – существенная, в противном случае - фиктивная.

Если обнаружено, что аргумент $\boldsymbol{x_i}$ фиктивный, то его можно исключить из рассмотрения, после чего таблица, задающая ФАЛ упрощается.

Необходимо исследовать существенность всех аргументов ФАЛ, описывающих заданный конечный автомат. При обнаружении фиктивных аргументов ФАЛ следует составить для нее новую таблицу истинности, не содержащую фиктивных аргументов.

Для ФАЛ, не содержащей фиктивных аргументов, должны быть образованы СДНФ и СКНФ. Полученные функции упростить, используя операции «поглащения» и «склеивания».

Аналитическое представление ФАЛ.

Минтермом ранга n $F_j^{(n)}$ называется n-местная ФАЛ в форме логического произведения (конъюнкции) n логических переменных *($x_1, x_2, \ldots, x_n$)*, в которое каждая переменная входит один раз в прямой или инверсной форме. Здесь j – номер минтерма.

Число различных минтермов n-го ранга, отличающихся индексом j , равно 2^n. Вхождению переменной x_i в прямой форме в минтерм соответствует 1, а в инверсной форме – 0. Десятичному номеру j соответствует n-разрядный двоичный набор.

Например: Минтерм 5-го ранга: $F_{19}^{(5)} = x_1\bar{x}_2\bar{x}_3x_4x_5$. J =19 = <10011>.

Свойства минтермов:

1. Любой минтерм n-го ранга равен 1 на одном единственном наборе, соответствующем номеру минтерма в n-разрядной двоичной записи, и равен 0 на всех других наборах.

Например, $F_{19}^{(5)} = x_1\bar{x}_2\bar{x}_3x_4x_5 = \begin{cases} 1 \quad \text{на} \quad \text{наборе} \quad <10011>=19; \\ 0 \quad \text{на} \quad \text{всех} \quad \text{других} \quad \text{наборах.} \end{cases}$

2. Логическая сумма (дизъюнкция) всех (!) минтермов n-го ранга равна 1, т.е.

$$\bigvee_{j=0}^{2^n-1} F_j^{(n)} = 1.$$

Алгоритм построения СДНФ таблично заданной ФАЛ:

1. Образовать множество T_1 – наборов, на которых функция принимает значение 1.

2. Выписать все минтермы максимального ранга, соответствующие наборам из T_1.

3. Объединить полученные минтермы знаком дизъюнкции.

Представление n-местной ФАЛ в форме дизъюнкции всех минтермов максимального (n-го) ранга, равных 1 на наборах из множества T_1, называется ***совершенной дизъюнктивной нормальной формой*** ***(СДНФ)***. Представление ФАЛ в виде дизъюнкции минтермов, не обязательно максимального ранга, называется **дизъюнктивной нормальной формой (ДНФ).**

Макстермом ранга n $\Phi_j^{(n)}$ называется n-местная ФАЛ в форме логической суммы (дизъюнкции) n логических переменных *($x_1,x_2,...,x_n$)*, в которую каждая переменная входит один раз в прямой или инверсной форме. Здесь j – номер макстерма.

Число различных макстермов n-го ранга, отличающихся индексом j , равно 2^n. Вхождению переменной x_i в прямой форме в макстерм соответствует 1, а в инверсной форме – 0. Десятичному номеру j соответствует n-разрядный двоичный набор.

Например: Макстерм 5-го ранга: $\Phi_{19}^{(5)} = x_1 \vee \bar{x}_2 \vee \bar{x}_3 \vee x_4 \vee x_5$; J =19 = <10011>.

Свойства макстермов:

I. Любой макстерм n-го ранга $\Phi_j^{(n)}$ равен **0** *на* одном единственном *наборе*, *противоположном* двоичной записи номера макстерма, имеющем десятичный номер **(2^n–1–j)**, и равен 1 на всех других наборах.

Например, $\Phi_{19}^{(5)} = x_1 \vee \bar{x}_2 \vee \bar{x}_3 \vee x_4 \vee x_5$

равен 0 на наборе J = < $\bar{0}\,\bar{1}\,\bar{0}\,\bar{0}\,\bar{1}$> = ($2^5$ -1) -j = <10011> = (32 – 1)-19 = 12, и

равен 1 на всех остальных наборах.

2. Логическое произведение (конъюнкция) всех (!) макстермов n-го ранга равна 0, т.е.

$$\mathop{\&}_{j=0}^{2^n-1} \Phi_j^{(n)} = 0.$$

Представление n-местной ФАЛ в форме конъюнкции макстермов максимального (n-го) ранга, равных 0 на наборах их множества T_0, называется **совершенной конъюнктивной нормальной формой (СКНФ).**:

Алгоритм построения СКНФ таблично заданной ФАЛ:

1. Образовать множество T_0 – наборов, на которых функция принимает значение 0.

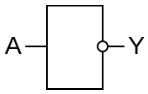
2. Выписать все макстермы максимального ранга, равные 0 на наборах из T_0.

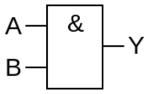
3. Объединить полученные макстермы знаком дизъюнкции.

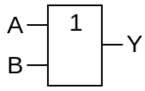
Представление ФАЛ в виде конъюнкции макстермов, не обязательно максимального ранга, называется **конъюнктивной нормальной формой(КНФ)**;

Комбинационные (переключательные) **схемы**

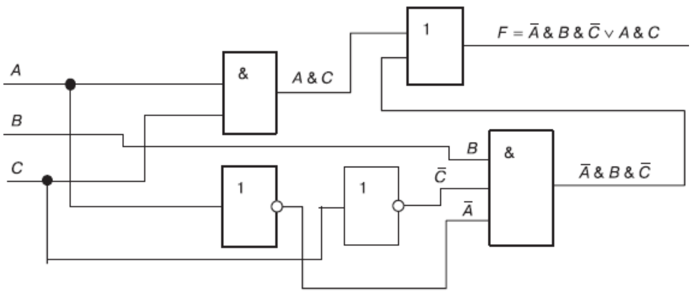
Для технической реализации ФАЛ используются, так называемые комбинационные (переключательные) схемы, которые составляются, обычно, с помощью элементов, реализующих стандартные логические функции «И», «ИЛИ», «НЕ» (см. рис.).

«НЕ» 

«И» 

«ИЛИ» 

Пример схемы



Качество схемы оценивается числом **Q** входящих в нее элементов, которое называется сложностью схемы. Схема, для которой сложность Q принимает наименьшее значение, называется минимальной (оптимальной).

Контрольные вопросы:

1. Сколько существует различных n-местных ФАЛ?
2. Какие аргументы называются фиктивными?
3. Каков алгоритм проверки аргумента на фиктивность?
4. Что такое СДНФ, ДНФ (СКНФ, КНФ)?
5. Как построить СДНФ, СКНФ?
6. Каковы свойства элементарных ФАЛ?
7. В чем суть операции склеивания и поглощения?
8. Что такое комбинационные схемы?
9. Как определяется сложность комбинационной схемы?

Литература

1. Булыгин В.С. Логические основы теории дискретных устройств: учеб. пособие. - М.: МАИ, 1983.

2. Булыгин В.С., Ескин В.И. Лабораторные работы «Дискретная математика (логические функции, конечные автоматы, графы)», — М.: МАИ, 2011.

Пример задания из ws-dss:

Метод: test_bool
Пользователь:
Входные данные:

```
{"team":_________}
```

Выходные данные:

Уникальный номер бригады _______________

Упростить, составить таблицу истинности, записать изображающие числа:

0. $f(x_0,x_1,x_2,x_3)=(((x_1\leftrightarrow x_0)\vee(x_1\downarrow x_3))\oplus((x_0|x_1)\vee(x_3\leftrightarrow x_1)))\oplus((\overline{x_1}\wedge(x_3\leftrightarrow x_2))\leftrightarrow((x_0\vee x_0)\oplus(x_1\rightarrow x_3)))$f(x0,x1,x2,x3)=(((x1↔️x0)∨(x1↓x3))⊕((x0|x1)∨(x3↔️x1)))⊕((x1¯∧(x3↔️x2))↔️((x0∨x0)⊕(x1→x3)))
1. $f(x_0,x_1,x_2,x_3)=((\overline{x_0}\wedge x_3)\vee\overline{(x_0\vee x_1)})\rightarrow(\overline{(x_3\rightarrow x_1)}\vee\overline{(x_0\downarrow x_1)})$f(x0,x1,x2,x3)=((x0¯∧x3)∨(x0∨x1)¯)→((x3→x1)¯∨(x0↓x1)¯)
2. $f(x_0,x_1,x_2,x_3)=x_3\rightarrow((x_2|(x_3|x_3))\vee x_2)$f(x0,x1,x2,x3)=x3→((x2|(x3|x3))∨x2)
3. $f(x_0,x_1,x_2,x_3)=(x_0\vee(\overline{x_1}\wedge(x_1\leftrightarrow x_3)))\vee(((x_3|x_1)|x_0)\oplus((x_1\downarrow x_0)\wedge(x_2\wedge x_1)))$f(x0,x1,x2,x3)=(x0∨(x1¯∧(x1↔️x3)))∨(((x3|x1)|x0)⊕((x1↓x0)∧(x2∧x1)))
4. $f(x_0,x_1,x_2,x_3)=(((x_1\downarrow x_2)\rightarrow(x_3\oplus x_3))\vee((x_1\downarrow x_0)\oplus\overline{x_0}))|(((x_0\wedge x_3)\leftrightarrow\overline{x_2})\oplus((x_1\rightarrow x_1)\wedge(x_3\vee x_2)))$f(x0,x1,x2,x3)=(((x1↓x2)→(x3⊕x3))∨((x1↓x0)⊕x0¯))|(((x0∧x3)↔️x2¯)⊕((x1→x1)∧(x3∨x2)))

Код ошибки/проверки: 0/0/0
Создан: 2018-09-06 12:26:14 +0300

<u>Здесь знак инверсии относится</u> к предыдущей переменной или к предыдущему выражению в скобках.

Метод: blackbox
Метод: blackbox
Пользователь: _______
Входные данные:

```
{
  "team": 20190310700
}
```

Выходные данные:

```
Дана ФАЛ:
x1| x2| x3| x4| x5| y
 0|  0|  0|  0|  0|  0
 0|  0|  0|  0|  1|  1
 0|  0|  0|  1|  0|  1
 0|  0|  0|  1|  1|  0
```

```
 0|  0|  1|  0|  0|  1
 0|  0|  1|  0|  1|  0
 0|  0|  1|  1|  0|  0
 0|  0|  1|  1|  1|  0
 0|  1|  0|  0|  0|  0
 0|  1|  0|  0|  1|  1
 0|  1|  0|  1|  0|  1
 0|  1|  0|  1|  1|  0
 0|  1|  1|  0|  0|  1
 0|  1|  1|  0|  1|  0
 0|  1|  1|  1|  0|  0
 0|  1|  1|  1|  1|  0
 1|  0|  0|  0|  0|  0
 1|  0|  0|  0|  1|  1
 1|  0|  0|  1|  0|  0
 1|  0|  0|  1|  1|  1
 1|  0|  1|  0|  0|  0
 1|  0|  1|  0|  1|  1
 1|  0|  1|  1|  0|  1
 1|  0|  1|  1|  1|  1
 1|  1|  0|  0|  0|  0
 1|  1|  0|  0|  1|  1
 1|  1|  0|  1|  0|  0
 1|  1|  0|  1|  1|  1
 1|  1|  1|  0|  0|  0
 1|  1|  1|  0|  1|  1
 1|  1|  1|  1|  0|  1
 1|  1|  1|  1|  1|  1


Найти фиктивные аргументы.
Указать любой один совпадающий набор для всех существенных аргументов.
В наборе перечислить значения всех аргументов кроме существенного.
Если совпадающих наборов нет, то указать []
```