МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный морской технический университет» (СПбГМТУ)

Кафедра Судовой Автоматики и Измерений

СИСТЕМЫ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ

Методические указания к выполнению курсового проекта по дисциплине Синтез логических систем.

ВЗФ

Санкт-Петербург 2014

Курсовой проект по дисциплине Синез логических систем выполняется студентами вечернего и заочного образования на 4 году обучения. Проект выполняется по индивидуальным заданиям. Представляется в виде пояснительной записки, сродержащей расчетную, а также функциональные и принципиальные схемы логических блоков в соответствии с заданием.

Ниже изложены основные сведения из теории и рекомендации по работе над составными частями проекта.

Первая часть проекта

1. По таблице истинности составить логические уравнения для каждого выхода в виде СДНФ и СКНФ.

Совершенная дизьюнктивная логическая форма (СДНФ) представляется логической суммой простых конъюнкций, каждая из которых содержит все переменные в прямом или инверсном виде не более одного раза; в такие конъюнкции не входят суммы переменных, а также отрицания произведений двух переменных или более. Входящие в СДНФ конъюнкции называются минтермами или конституентами единиц. Например:

$$F=\bar a\bar b\bar c\bar d+\bar a\bar b\bar c d+\bar a\bar b c\bar d+\bar a b\bar c\bar d+\bar a bc\bar d+a\bar b\bar c\bar d+$$
$$+a\bar b\bar c d+a\bar b c\bar d$$

Формула получается в два этапа:

а) Количество слагаемых равно числу наборов таблицы истинности, на которых логическая функция равна «1», а каждое слагаемое это логическое произведение всех независимых переменных

б) ставится знак инверсии над теми независимыми переменными, которые равны «0» в рассматриваемом наборе (строке таблицы).

Совершенная коньюктивная нормальная форма (СКНФ) представляется логическим произведением дизъюнкций, каждая из которых содержит все переменные в прямом или инверсном виде не более одного раза. Входящие в произведение сомножители – дизъюнкции – называются макстермами или конституентами нулей. Например:

$$F=(a+b+\bar{c}+\bar{d})\cdot(a+\bar{b}+c+\bar{d})\cdot(a+\bar{b}+\bar{c}+\bar{d})\cdot(\bar{a}+b+\bar{c}+\bar{d})\cdot$$
$$(\bar{a}+\bar{b}+c+d)\cdot(\bar{a}+\bar{b}+c+\bar{d})\cdot(\bar{a}+\bar{b}+\bar{c}+d)\cdot(\bar{a}+\bar{b}+\bar{c}+\bar{d})$$

Формула также получается в два этапа:

а) количество сомножителей равно числу наборов таблицы истинности, на которых логическая функция равна «0», а каждый сомножитель это логическая сумма всех независимых переменных;

б) ставится знак инверсии над теми независимыми переменными, которые равны «1» в рассматриваемом наборе.

Структурные формулы в виде СДНФ и СКНФ эквивалентны и, с помощью законов алгебры, логики могут быть преобразованы одна в другую.

2. Для получения наиболее простой логической схемы выполнить минимизацию функций, записанных в СДНФ, используя метод непосредственных преобразований.

Минимизацией называют процедуру упрощения логической функции, с тем чтобы она содержала минимальное количество членов при минимальном числе переменных. При этом используются законы алгебры логики.

Законы Булевой алгебры:

Ниже приведены основные законы, применяемые при синтезе комбинационных схем.

Закон перестановки: $a \times b = b \times a$ и $a + b = b + a$.

Ассоциативный закон: $a \times (b \times c) = (a \times b) \times c$ и $a + (b + c) = (a + b) + c$.

Дистрибутивный закон: $a \times (b + c) = a \times b + a \times c$ и $a + (b \times c) = (a + b) \times (a + c)$.

Закон двойного отрицания: $\bar{\bar{a}} = a$.

Тавтологии: $a \times a = a$ и $a + a = a$.

Закон нулевого элемента: $a \times 0 = 0$ и $a + 0 = a$.

Закон единичного элемента: $a \times 1 = a$ и $a + 1 = 1$.

Закон дополнительного элемента: в Булевой алгебре дополнительным элементом к a является $\bar{a}$, поэтому: $a + \bar{a} = 1$ и $a \times \bar{a} = 0$.

Закон двойственности (де Моргана): $\overline{a \times b} = \bar{a} + \bar{b}$ и $\overline{a + b} = \bar{a} \times \bar{b}$.

Как из этого закона вытекает: $a \times b = \overline{\bar{a} + \bar{b}}$ и $a + b = \overline{\bar{a} \times \bar{b}}$.

Закон: $a + a \times b = a$ и $a \times (a + b) = a$.

Закон: $a + \bar{a} \times b = a + b$ и $a \times (\bar{a} + b) = a \times b$.

Как из этого закона вытекает: $\bar{a} + a \times b = \bar{a} + b$ и $\bar{a} \times (a + b) = \bar{a} \times b$.

Закон: $a \times b + a \times \bar{b} = a$ и $(a + b) \times (a + \bar{b}) = a$.

Например, пусть F равно

$$\mathrm{F}=\bar{a}\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}\bar{c}d + \bar{a}\bar{b}c\bar{d} + \bar{a}b\bar{c}\bar{d} + \bar{a}bc\bar{d} + a\bar{b}\bar{c}\bar{d} +$$
$$+a\bar{b}\bar{c}d + a\bar{b}c\bar{d}$$

Получим:

$$1 + 3 : \bar{a}\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}c\bar{d} = \bar{a}\bar{b}\bar{d}$$
$$3 + 8 : \bar{a}\bar{b}c\bar{d} + a\bar{b}c\bar{d} = \bar{b}c\bar{d}$$
$$2 + 7 : \bar{a}\bar{b}\bar{c}d + a\bar{b}\bar{c}d = \bar{b}\bar{c}d$$
$$4 + 5 : \bar{a}b\bar{c}\bar{d} + \bar{a}bc\bar{d} = \bar{a}b\bar{d}$$
$$1 + 6 : \bar{a}\bar{b}\bar{c}\bar{d} + a\bar{b}\bar{c}\bar{d} = \bar{b}\bar{c}\bar{d}$$

Дальше склеиваем 1+4 и 2+5 и окончательно

$$\mathrm{F} = \bar{\mathrm{a}}\bar{d} + \bar{b}\bar{c} + \bar{b}\bar{c}d$$

Следует отметить, что элементарные приемы минимизации удаётся использовать не часто – при малом количестве членов функции и небольшом числе переменных. В других случаях применяются специальные методы минимизации, облегчающие поиск склеивающихся членов, например, метод минимизации с помощью карт Карно.

3. Привести полученные минимизированные функции к базису И-НЕ. Применяя закон де Моргана получим

$$\mathrm{F} = \overline{\overline{\bar{\mathrm{a}}\bar{d} + \bar{b}\bar{c} + \bar{b}\bar{c}d}} = \overline{\overline{\bar{a}\bar{d}} \cdot \overline{\bar{b}\bar{d}} \cdot \overline{\bar{b}\bar{c}d}}$$

4. Выполнить минимизацию функций с помощью карт Карно и сравнить полученные результаты.

Карта Карно построена так, что в её соседние клетки попадают смежные члены функции – члены, отличающиеся значением одной переменной: в один член эта переменная входит в прямой форме, а в другой – в инверсной. Благодаря этому возникает наглядное представление о различных вариантах смежных членов.

Карта Карно имеет столько клеток, сколько комбинаций (наборов) можно составить из прямых и инверсных значений n переменных по n членов в каждой. Так, при n = 2 карта содержит четыре клетки, при n = 3 – восемь клеток, при n = 4 – шестнадцать клеток.

Наборы переменных, на которых y = 1, т.е. минтермы функции, отмечаются в соответствующих клетках карты единицами, в остальные клетки записываются нули или их оставляют пустыми. Две стоящие в соседних клетках единицы – свидетельство того, что в составе СДНФ имеются члены, отличающиеся значением одной переменной. Такие члены склеиваются. Склеивание каждой пары минтермов уменьшает число входящих в них переменных на единицу.

Общие правила склеивания членов, занесённых в карту Карно следующие:

1) склеиваться могут 2, 4, 8, … членов; при этом соответствующие единицам клетки для наглядности охватывают контурами; каждый должен быть прямоугольником;

2) одним контуром следует объединять максимальное количество клеток;

3) одна и та же единица может охватываться разными контурами, т.е. один и тот же минтерм может склеиваться с несколькими смежными; последнее объясняется тем, что значение функции не меняется при прибавлении уже имеющихся членов;

4) крайние строки, а также крайние столбцы карты считаются смежными. Функция, минимизированная с помощью карты Карно, состоит из суммы простых конъюнкций. Каждая из них получается в результате склеивания членов, которым соответствует охваченные контуром единицы. В такую конъюнкцию войдут только те переменные, значения которых в пределах контура не меняются.

Карта Карно для F:

$$F=\bar a\bar b\bar c\bar d+\bar a\bar b\bar c d+\bar a\bar b c\bar d+\bar a b\bar c\bar d+\bar a b c\bar d+a\bar b\bar c\bar d+$$

$$+a\bar b\bar c d+a\bar b c\bar d$$

cd		ab 00	01	11	10
	00	1(1)	1(4)		1(6)
	01	1(2)			1(7)
	11				
	10	1(3)	1(5)		1(8)

Рис. 1. Карта Карно

$$F=\overline{\overline{\bar a\bar d+\bar b\bar c+\bar b\bar c d}}=\overline{\overline{\bar a\bar d}\cdot\overline{\bar b\bar c}\cdot\overline{\bar b\bar c d}}$$

5. Выбор аппаратных средств.

Для реализации этой функции потребуется:

4 эл-та 2И-НЕ - инвертирование входных переменных

2 эл-та 2И-НЕ - первый и второй сомножители

1 эл-т 3И-НЕ - третий сомножитель

1 эл-т 3И-НЕ – завершающая конъюнкция

Функциональная схема изображена на рисунке 2.

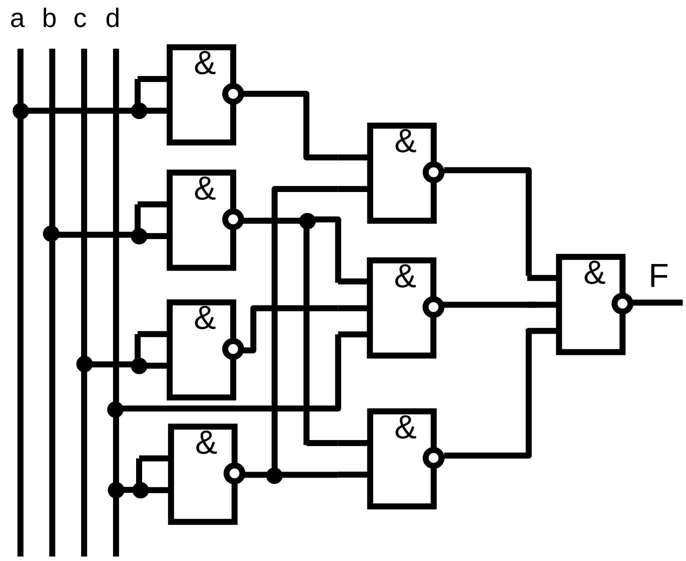

Рис. 2. Функциональная схема комбинационного логического блока

6. Пример выполнения принципиальной схемы и перчня элементов приведен в указаниях к третьей части проекта.

Вторая часть проекта

Коды, используемые в курсовом проекте, является комплектными. Это означает, что все комбинации данного кода содержат одинаковое количество двоичных разрядов.

Основные сведения о кодах и построении их кодовых комбинаций приводятся ниже.

1. Трехразрядный код на все сочетания.

Это обычный двоичный код, каждая кодовая комбинация которого содержит 3 разряда. Общее количество различных кодовых комбинаций при этом равно 2^3 =8. Этот код является примером арифметических кодов, т.е. таких кодов, в основе построения которых лежат известные системы счисления.

2. Код Грея.

Этот код относится к классу специальных кодов, носящих название отраженных или рефлексных. Отличительной особенностью этих кодов является то, что соседние кодовые комбинации различаются между собой цифрой только в одном разряде. Это позволяет уменьшить ошибки неоднозначности считывания цифровой информации.

Любая кодовая комбинация в коде Грея может быть получена из соответствующей кодовой комбинации кода на все сочетания, если выполнить следующие операции:

а) под комбинацией исходного кода записать такую же комбинацию, но сдвинутую на один разряд вправо (при этом младший разряд сдвигаемой комбинации отбрасывается);

б) произвести поразрядное сложение сдвинутой и несдвинутой кодовых комбинаций по модулю два.

Правила сложения по модулю два следующие:

$0 \oplus 0 = 0,$

$0 \oplus 1 = 1,$

$1 \oplus 0 = 1,$

$1 \oplus 1 = 0$

В качестве примера преобразуем кодовую комбинацию кода на все сочетания 101 в код грея. Для этого выполним указанные выше действия:

101 - исходная комбинация

  101 - сдвинутая на разряд вправо исходная комбинация

111 - комбинация кода Грея

Так преобразуют все кодовые элементы исходного кода.

3. Двоично-десятичный код 2 - 4 - 2 -1 (ДДК-2421)

Этот код (иначе он называется кодом Айкена) широко используется в ЭВМ и служит для преобразования десятичного числа в двоичный эквивалент. При этом каждому разряду десятичного числа ставится в соответствие четыре разряда (тетрада) двоично-десятичного кода, веса которых, начиная со старшего, соответственно равны 2,4,2,1. Каждый разряд десятичного числа преобразуется в соответствующую тетраду независимо друг от друга. Например, десятичному числу 27 соответствует комбинация 0010 1101 в двоично-десятичном коде. Таким образом, любая кодовая комбинация двоично-десятичного кода имеет количество двоичных разрядов, кратное четырем, и определяемое разрядностью преобразуемого десятичного числа.

При образовании комбинаций кода 2 - 4 - 2 -1 следует учитывать его свойство самодополняемости. Оно состоит в том, что первые десятичные цифры (0,1,2,3,4) соответствуют, обычному двоичному коду, 5 - это 1011. Для большей цифры двоично-десятичный код находится как инверсия двоичных кодов десятичного числа, дополняющего исходное до 9.

Например. Необходимо определить кодовую комбинацию ДДК 2 - 4 - 2 - 1, соответствующую цифре 6. Дополнением к 6 до 9 является 3, а в двоичном коде 0011. Выполняем простую инверсию разрядов и получаем 1100.

4. Двоично-десятичный код 8 - 4 -2 - 1

Принципиально этот код не отличается от кода 2 - 4 - 2 - 1. Отличие состоит лишь в том, что веса разрядов, стоящих в каждой тетраде имеют другие значения. Это означает, по существу, что каждая тетрада в коде 8 - 4 - 2 - 1 формируется по правилу двоичного четырехразрядного кода на все сочетания. Например, десятичному числу 38 соответствует комбинация двоично-десятичного кода 0011 1000.

5. Код Джонсона.

Этот код также относится к классу двоично-десятичных. Здесь каждому разряду десятичного числа соответствует комбинация из пяти двоичных разрядов, в которой число единиц, начиная с младшего разряда, для чисел от 0 до 5 возрастает на единицу с увеличением цифры десятичного числа, а для чисел, больших 5, - уменьшается на единицу. Так, цифре 3 соответствует комбинация 00111, а цифре 7-11100.

6. Семисегментный десятичный код.

Этот код служит для "высвечивания" десятичной цифры на цифровом индикаторе, состоящем из семи отдельных элементов (сегментов). Эти сегменты пронумерованы (Рис. 3).

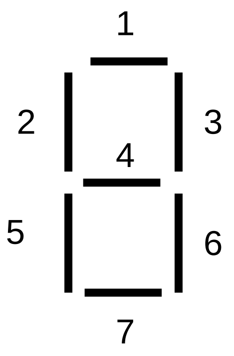

Рис. 3. Семисегментный индикатор

При подаче напряжения на какой-либо элемент, он "высвечивается". Таким образом, для воспроизведения десятичной цифры необходимо подать напряжение на соответствующие сегменты, из которых и появляется стилизованное изображение цифры. Например, для цифры 2 необходимо подать напряжение на сегменты 1,3,4,5,7.

7. Код с постоянным весом C_n^m

Этот код является примером кода с обнаружением ошибок. Каждое кодовое слово длинной *n* (содержащее *n* двоичных разрядов) содержит *m* единиц, остальные - нули.

Общее число разрешенных кодовых комбинаций в двоичном коде с постоянным весом равно

$$N = C_n^m = \frac{m!}{m!(n-m)!}$$

Нетрудно видеть, что вообще возможно образовать 2^n различные n-разрядные двоичные кодовые комбинации. Из них только N будут содержать *m* единиц, остальные (2^n - *N)* имеют число единиц, отличное от *m*, считаются запрещенными, и не используются. Проектируемая комбинационная схема должна фиксировать отклонение числа единиц во входной кодовой комбинации от *m*.

8. *n*-разрядный код с проверкой на четность.

В таком коде из 2^n возможных кодовых комбинаций разрешенными считаются только те, которые содержат четное число единиц. Таким образом, число разрешенных комбинаций равно 2^{n-1}, то есть в два раза меньше. В курсовом проекте предлагается разработать схему защиты много разрядного кода обнаруживающую и запрещающую коды, содержащие нечетное число единиц.

9. *n*-разрядный код с проверкой на нечетность.

Этот код принципиально не отличается от кода с проверкой на четность. Запрещенными здесь считаются комбинации, содержащие четное число единиц.

10. Распределительный 8-разрядный код.

Этот код является разновидностью кода с постоянным весом равным единице, т.е. при *m* =1. В любой разрешенной кодовой комбинации длиной *n* (в данном случае n =8) содержится только одна единица. Число таких кодовых комбинаций равно N= C_n^m = n. Разрешенные кодовые комбинации отличаются друг от друга местом расположения единицы.

Последовательность работы над заданием

А. Составление таблицы состояний и аналитическая запись функций алгебры логики.

Выпишем слева в таблице все возможные комбинации входного кода, а справа - соответствующие им комбинации выходного кода.

Введем обозначения для разрядов входной комбинации a,b,c,... и выходной $z1,\ z2,\ z3$..... Отметим, что в случае, когда речь идет о разработке схемы зашиты кода, выходная комбинация состоит го из одного разряда.

Тогда таблица состояний будет иметь вид, типичный фрагмент которой в случае преобразования входного трехразрядного кода 010 в семисегментный представлен в таблице 1 (нижняя строка). Как отмечено выше чтобы высветить цифру 2 необходимо зажечь сегменты 1,3,4,5,7.

Таблица 1

a	b	c	Цифра	Z7	Z6	Z5	Z4	Z3	Z2	Z1
...	...	...	0	...	...	...	...	...	...	...
...	...	...	1	0	...	...	1	1		
0	1	0	2	1	0	1	1	1	0	1

По таблице состояний логическая функция для каждого разряда выходной комбинации может быть записана в аналитическом виде в одной из совершенных нормальных форм, например, в СДНФ - совершенной дизъюнктивной нормальной форме.

Б. Минимизация.

Следующей задачей синтеза логических схем является упрощение выражении и нахождения минимальных дизъюнктивных нормальных форм (МДНФ). Методы минимизации рассмотрены в рекомендациях к первой части работы. Здесь рассмотрим дополнительно методику получения МДНФ по импликантной матрице. Выбор методики минимизации предоставляется исполнителю.

Этот метод предполагает выполнение следующих операций:

1. логическую функцию представляют в СДНФ;

2. в полученной СДНФ проводят все операции неполного склеивания и поглощения. В результате получается сокращенная дизъюнктивная нормальная форма, т.е. дизъюнкция самых коротких из всех возможных элементарных произведений (простых импликант), входящих в данную логическую функцию;

3. находят МДНФ по импликантной матрице.

Последняя представляет собой таблицу, на вертикальные и горизонтальные входы которой записываются соответственно члены СДНФ и простые импликанты заданной логической функции. Клетки импликантной матрицы, образованные пересечением строк с импликантам и столбцов с поглощаемыми ими членами СДНФ, каким-либо образом помечаются, например, крестиками.

МДНФ находится как дизъюнкция минимального числа имилнкант, которые совместно накрывают крестиками все столбцы импликактной матрицы.

В качестве примера рассмотрим получение МДНФ с помощью таблицы импликант для логической функции z, записанной в СДНФ:

$$z = abc + ab\bar{c} + a\bar{b}c + \bar{a}\bar{b}c + \bar{a}\bar{b}\bar{c}$$

Производя все операции неполного склеивания, получим следующие импликанты: $ab, ac, \bar{b}c, \bar{a}\bar{b}$

Составляем импликантную матрицу (табл. 2).

Таблица 2

Импликантная матрица для функции z

	abc	$ab\bar{c}$	$a\bar{b}c$	$\bar{a}\bar{b}c$	$\bar{a}\bar{b}\bar{c}$
ab	+	+			
ac	+		+		
$\bar{b}c$			+	+	
$\bar{a}\bar{b}$				+	+

Из таблицы 2 видно, что можно получить две равноценные по количеству входящих в формулу букв МДНФ функции z; одна из них находится как дизъюнкция имнликант, отмеченных сплошной линией, и запишется в виде:

$z = ab + ac + \overline{a}\overline{b}$, а вторая из импликант, отмеченных штриховой линией:

$z = ab + \overline{b}c + \overline{a}\overline{b}$

В заключение отметим, что все рассмотренные выше способы минимизации не исключают на заключительном этапе дальнейшее упрощение логической функции путем вынесения общих множителей за скобки (если это возможно и целесообразно).

В. Составление функциональной схемы.

По полученным для каждого разряда выходной функции минимальным формам составляется функциональная схема разрабатываемого устройства.

Следует отметить, что каждый элемент схемы отражает лишь функциональную связь между входными и выходными переменными. Это, в частности, означает, что количество входных переменных у одного блока может быть сколь-угодно большим и каждый выход блока может нагружаться произвольным количеством нагрузок.

Составленная таким образом функциональная схема разрабатываемого устройства приводится на отдельном листе формата А4 в пояснительной записке. Требуется соблюдать стандарт на изображение логических элементов в соответствии с ГОСТ 2743-72 ЕСКД.

Пример функциональной схемы приведен на рисунке 2.

Третья часть проекта

Структура разрабатываемого автомата изображена на рисунке 4.

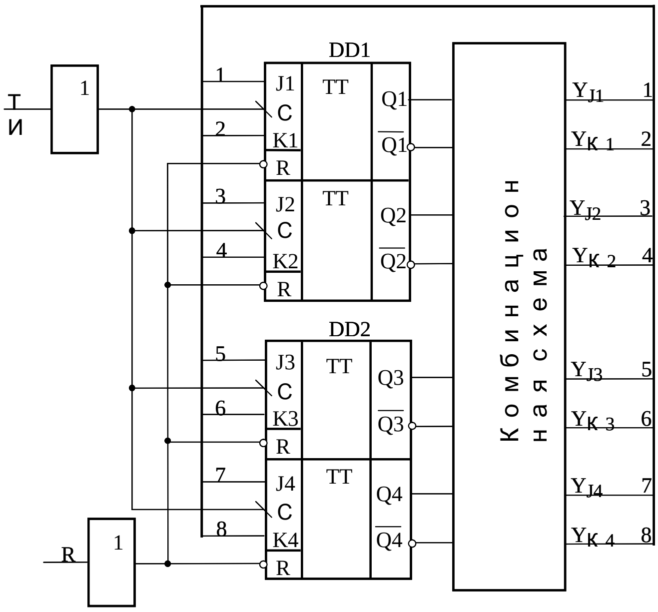

Рис. 4. Структура разрабатываемого автомата

Пусть заданием предусмотрена разработка автомата, имеющего K_{max}=11 и пропуск состояний N_1=4, N_2=8 и N_3=10. Составим таблицу истинности описывающую закон функционирования автомата (табл. 3).

Таблица 3

	Выходы триггеров				
№	Q_3	Q_2	Q_1	Q_0	
0	0	0	0	0	Начальное состояние
1	0	0	0	1	
2	0	0	1	0	
3	0	0		1	
4	–	–	–	–	Пропуск
5	0	1	0	1	
6	0	1	1	0	
7	0	1	1	1	
8	–	–	–	–	Пропуск
9	1	0	0	1	
10	–	–	–	–	
11	1	0	1	1	Конечное состояние
	Возвращение в нулевую строку				

Закон функционирования *JK*-триггера иллюстрируется сокращенной таблицей переходов (табл. 4). В таблице показаны все возможные переходы триггера из одного состояния в другое, в ней символом X обозначено безразличное состояние.

Под действием входных импульсов счетчик переходит из одного состояния в другое. Комбинация состояний выходов триггеров определяет двоичное число на выходе счетчика. Значение этого числа увеличивается на единицу или устанавливается равным нулю после достижения максимального значения *N*-1. Такие переходы из одного состояния в другое заносятся в таблицу переходов каждого триггера счетчика (табл. 4).

Таблица 4

J	K	Q_n	Q_{n+1}	
0	X	0	0	Хранение, Уст_0
1	X	0	1	Уст_1
X	1	1	0	Уст_0, Инверсия
X	0	1	1	Хранение, Уст_1

Для осуществления таких переходов на входах *J* и *K* каждого из триггеров должна быть установлена соответствующая комбинация управляющих сигналов J_n и K_n, необходимых для функционирования счетчика по заданному алгоритму.

Рассмотрим, последовательность действий по проектированию счетчика с K_{max} = 9 и пропуском значений K_1 = 1, K_2 = 3, K_3 = 5. Прежде всего, составим таблицу истинности (см. табл. 3).

На основании таблицы истинности заполним таблицу переходов триггеров T1 – T4 и управляющих сигналов J_i, K_i на их входах, обеспечивающих эти переходы. Методика заполнения таблицы по *J* и *K* входам следующая. Рассматривается *n*-ая строка для выходов триггеров Q1 – Q4 и в предыдущую строку *n*-1 для входов *J* и *K* вписываются те значения, которые обеспечивают переход триггеров из исходного состояния (строка *n*-1) в новое состояние (строку *n*). Например, при переходе от N=0 к N=2 триггер Q2 переходит из состояния *лог.*0 в состояние *лог.*1. В соответствии с таблицей 4 мы записываем J_2 = 1 и K_2 = X.

В результате описания всех переходов получаем таблицу функционирования синтезируемого счетчика.

Таблица 5

N	Q4	Q3	Q2	Q1	J_4	K_4	J_3	K_3	J_2	K_2	J_1	K_1
0	0	0	0	0	0	X	0	X	1	X	0	X
2	0	0	1	0	0	X	1	X	X	1	0	X
4	0	1	0	0	0	X	X	0	1	X	0	X
6	0	1	1	0	0	X	X	0	X	0	1	X
7	0	1	1	1	1	X	X	1	X	1	X	1
8	1	0	0	0	X	0	0	X	0	X	1	X
9	1	0	0	1	X	1	0	X	0	X	X	1

Минимизацию логических функций проведем с помощью карт Карно. Карта Карно для переменной J4 изображена на рисунке 5.

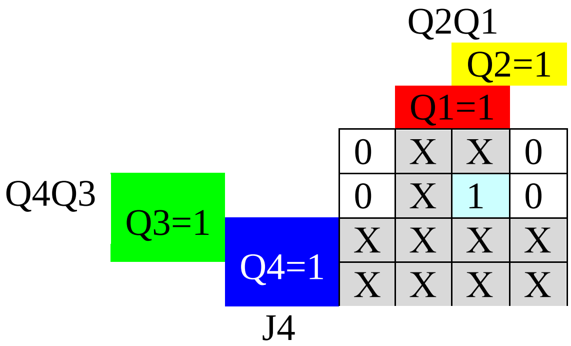

Рис. 5. Карта Карно для J4

Из карты получаем минимизированное логическое выражение для функции J4: $J4 = Q1$

Карты Карно для остальных 7 переменных J3 K4 … J1 K1 приведены на рисунке 6.

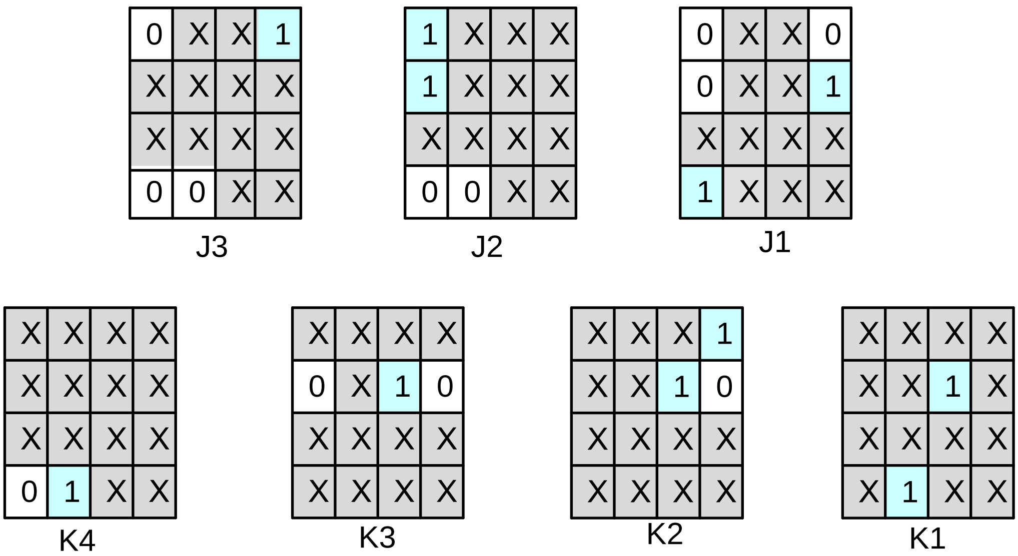

Рис. 6. Карты Карно для переменных J3-J1 и K4-K1

По картам Карно получим минимизированные выражения функций:

$$J1 = Q2 \cdot Q3 + Q4$$
$$J2 = \overline{Q4}$$
$$J3 = Q2$$
$$J4 = Q1$$

$$K1 = 1$$
$$K2 = \overline{Q3} + Q1$$
$$K3 = Q1$$
$$K4 = Q1$$

Полученные соотношения требуется записать в базисе И-НЕ. Для этого преобразуем выражение J1 по правилу де Моргана:

$$J1 = Q2 \cdot Q3 + Q4 = \overline{\overline{Q2 \cdot Q3} \cdot \overline{Q4}},$$
$$K2 = Q1 + \overline{Q3} = \overline{Q3 \cdot \overline{Q1}}.$$

Реализуем цифровой автомат на микросхемах серии К555. В этой серии имеется МС сдвоенного JK-триггера К555ТВ6. Для построения комбинационной схемы можно использовать двухвходовые МС 2И-НЕ (К555ЛА3). Для построения схемы (рис. 1) требуется два корпуса микросхем триггеров и один корпус МС К555ЛА3, содержащий 4 элемента 2И-НЕ (4х2И-НЕ).

Схема спроектированного счётчика представлена на рисунке 7. Схема вычерчивается вручную или с использованием технических средств на отдельном листе формата А4 и приводится в пояснительной записке. Схема сопровождается спецификацией (перечнем элементов).

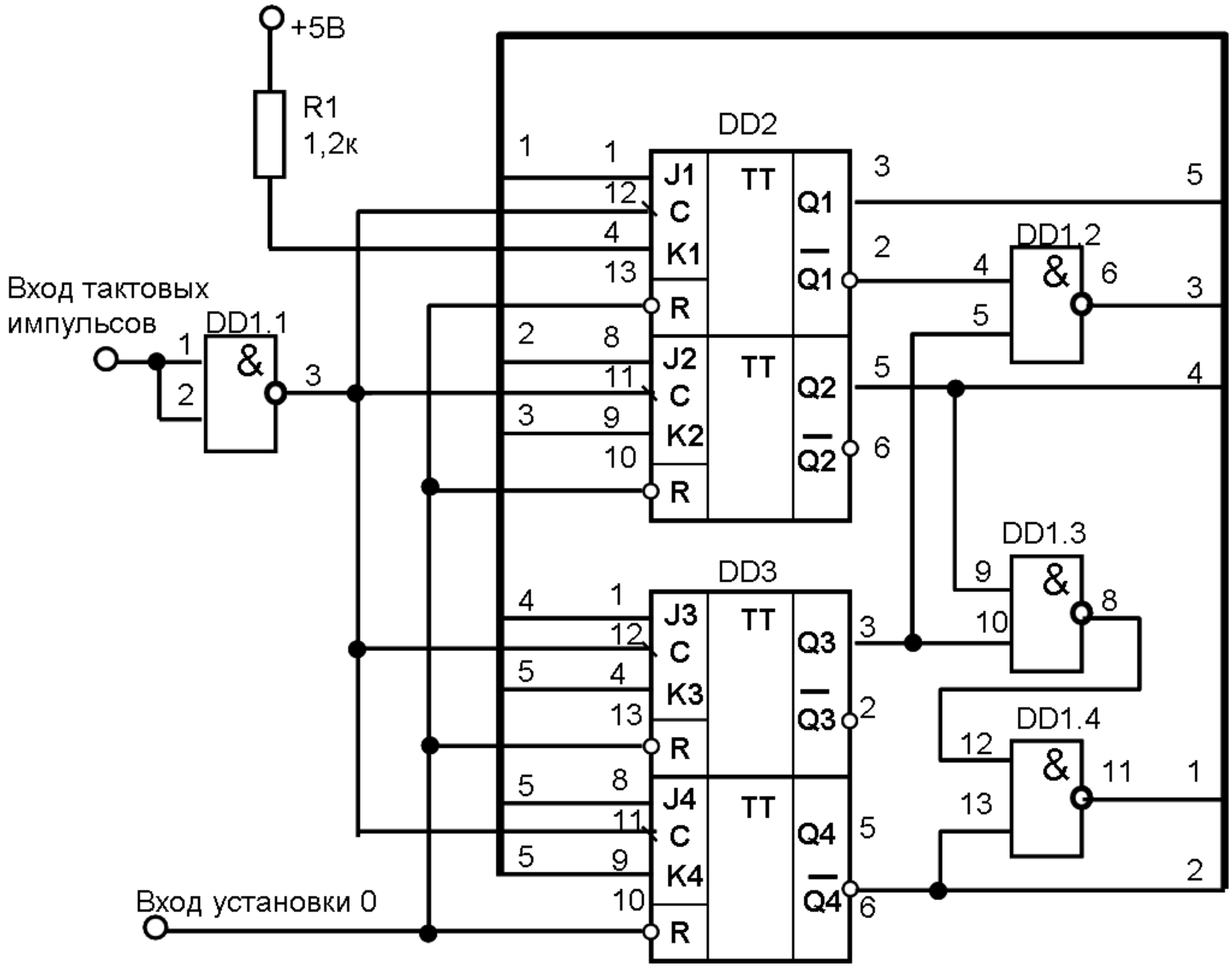

Рис. 7. Принципиальная схема счетчика

Спецификация элементов схемы Таблица 6

№	Обозначение	Наименование	Кол-во	Примечание (ТУ)
	Микросхемы			
1.	DD1	К555ЛА3	1	
	Резисторы			
4.	R1	Резистор С2 – 33Н - 0,125 – 1,2кОм - ±5%		АБШК.434110.046 ТУ

СОДЕРЖАНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Титульный лист (пример оформления в ПРИЛОЖЕНИИ 1)
Реферат (пример оформления в ПРИЛОЖЕНИИ 2)
Данные задания
ВВЕДЕНИЕ
ЧАСТЬ 1

- расчетная часть, содержащая логические уравнения для каждого выхода в виде СДНФ и СКНФ; минимизация выражений СДНФ и СКНФ методом непосредственных преобразований; приведение выражений функций к базису И-НЕ; минимизация с помощью карт Карно; сравнение результатов.
- функциональные схемы реализации функций;
- аппаратные средства, необходимые для реализации функций, как с использованием единого базиса, так и без использования единого базиса;
- выбор оптимального варианта и принципиальная схема со спецификацией;
- выводы.

ЧАСТЬ 2

- назначение комбинационной схемы
- расчетная часть, содержащая описание структуры входных и выходных сигналов; краткие сведения об используемых кодах, их особенностях, областях применения; таблица состояний; аналитические выражения логических функций; минимизация и обоснование метода.
- разработка функциональной схемы.
- выводы.

ЧАСТЬ3

- назначение схемы;
- расчетная часть, содержащая таблицы функционирования универсального триггера и счетчика, карты Карно; синтез принципиальной схемы; список используемой литературы;
- принципиальная схема и спецификация (перечень элементов);
- выводы

ЗАКЛЮЧЕНИЕ
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Нарышкин А.К. Цифровые устройства и микропроцессоры. Уч. пособие. М.: Академия. 2006 - 320с.
2. Пухальский Г.И. Новосельцева Т.я. Цифровые устройства. М.: Мир, 1996
3. Вирьянский Э.Я. Пректирование логических устройств судовой автоматики. Л.: Судостроение 1979г.
4. Шило В.Л. Популярные цифровые микросхемы: справочник. – М.: Радио и связь, 2002.
   http://www.bookam.net/book/shilo_v_l_/populjarnye_cifrovye_mikroshemy.html#
5. Справочник по интегральным микросхемам / Под ред. Тарабрина. - М.: Энергия, 1980, 1983
6. Нефедов А.В. Интегральные микросхемы и их зарубежные аналоги: Справочник. Т. 5, М.: КУбК-а, 1997. – 608с.

ПРИЛОЖЕНИЕ 1 - Титульный лист

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный морской технический университет» (СПбГМТУ)

Кафедра Судовой Автоматики и Измерений

СИСТЕМЫ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К КУРСОВОМУ ПРОЕКТУ
по дисциплине «Синтез логических систем»

Задание вариант № __

Выполнил студент гр. _____ ________________________ И.О. Фамилия
подпись, дата

Проверил ________________________ Н. В. Семидетнов
подпись, дата

Санкт-Петербург 201_

ПРИЛОЖЕНЕ 2 – Реферат (пример)

РЕФЕРАТ

Пояснительная записка к курсовому проекту по дисциплине "Синтез логических систем": __ страниц, __ рисунков, __ таблиц, __ источников.

Объектом разработки являются логические системы управления объектами автоматизации в соответствии с заданием на проектирование.

Цель работы – освоить методы анализа и синтеза логических систем управления, практически применить их и продемонстрировать умение разрабатывать функциональные и принципиальные схемы систем.

В пояснительной записке отражены этапы и существо работы по проектированию комбинационных и последовательностных логических блоков и устройств. Продемонстрировано умение составлять логические выражения, минимизировать их, приводить к заданному базису и разрабатывать принципиальные схемы.

Проект выполнен в соответствии с заданием и в полном объеме.