Лабораторная работа №6

**Анализ и синтез простейших схем комбинационной логики.**

***1. ЦЕЛЬ РАБОТЫ****.*

1. Составление таблицы истинности (ТИ) по результатам анализа цифрового устройства (ЦУ) с неизвестной схемой.

2. Синтез булевой функции (БФ), исходя из ТИ.

3. Реализация ЦУ на основе БФ с минимальными схемными затратами.

4. Верификация созданного ЦУ – проверка идентичности их ТИ.

***2. ПОДГОТОВКА К РАБОТЕ.***

***2.1****. Изучить следующие вопросы курса*:,

* составление ТИ по заданным условиям работы ЦУ,
* получение CДНФ из ТИ,
* минимизация СДНФ с помощью карт Карно.

***2.2****. Построение УКЛ на базе элементов EWB.*

При создании схем используются:

* логические элементы – палитра Logic Gates (Рис.6.1),
* контактные точки – палитра Basic,
* индикаторы – палитра Indicators (там же, где pA и pV).



Рис.6.1. Палитра базовых компонентов для синтеза цифровых схем.

При необходимости увеличения числа входов использовать *"Component properties → Number of inputs"*.

***!!!*** УГО элементов *ИЛИ (OR)* и *ИЛИ-НЕ (NOR)* в EWB отличается от общепринятого. Предъявляемый графический материал должен содержать *только стандартные обозначения* логической функции. На Рис.6.2 показано УГО для ИЛИ-НЕ



Рис.6.2. Неправильное и правильное УГО схемы 2ИЛИ-НЕ.

***2.3****. Основные свойства БФ и методы их преобразования в схему ЦУ.*

***2.3.1.*** *Задачи минимизации.*

Преобразование "ТИ → СДНФ" полагается известным.

Схемная реализация непосредственно СДНФ приводит к неоправданно большому количеству корпусов ИМС с различными функциями и соединений между ними. При создании схемы ЦУ необходимо учитывать, что:

* ИМС производят конкретные логические операции: НЕ, И, И-НЕ, ИЛИ, ИЛИ-НЕ, исключающее ИЛИ" (XOR), "исключающее ИЛИ-НЕ" (XNOR),
* каждый корпус ИМС содержит несколько независимых ЛЭ с одинаковой логической функцией, независимо от самой функции,

Основная задача – создать схему с наименьшим количеством корпусов ИМС. При неоднозначном решении лучшим считается то, которое предусматривает меньшее количество типов ИМС.

*Пример, актуальный для данной Л.Р.*

Решение может предусматривать два корпуса И-НЕ и ИЛИ-НЕ. Другое решение этой же задачи предусматривает тоже два корпуса, но оба И-НЕ; оно конечно, лучше.

***2.3.2.*** *Этапы минимизации.*

Преобразование "БФ → схема" содержит в общем случае три этапа:

* формальная минимизация СДНФ с помощью карт Карно имеет общие правила для любого вида СДНФ, получаемого из ТИ; есть в курсе лекций,
* минимизация с использованием преобразований де Моргана или выявления операций типа NOR и XNOR,
* минимизация для сокращения типов логических операций в конечной БФ, которая непосредственно предназначена для схемной реализации

***2.3.3****. Преобразования (теорема, формула) де Моргана.*

Имеет два равноправных вида

 (6.1)

Преобразование используется и для переменных составного вида, например,

 (6.2)

Выгода очевидна даже в (6.1): вместо двух типов операций (корпусов) – инверсия с сложение/умножение используется только один И-НЕ или ИЛИ-НЕ.

Еще более выгоден вариант в (6.2), когда один ЛЭ заменяет три операции.

***2.3.4****. Использование операций XOR и XNOR.*

Булевы функции операций

 (6.3)

Здесь также возможен составной вид переменных, например

 (6.4)

Эти функции имеют свою область применения, но в данном случае важно то, что имеется корпус 4-мя ЛЭ и каждый ЛЭ сразу производит операцию XOR/XNOR.

***!!!*** Вообще-то стандартно выпускается только XOR, но раз в палитре Logic Gates имеется XNOR, то можно им воспользоваться.

Выгода от использования очевидна из Рис.6.3.



Рис.6.3. Реализация XOR: a) "as is", b) на стандартном ЛЭ.

***2.3.5.*** *Сокращение типов логических операций*.

Каждый корпус содержит четыре ЛЭ, т.е. некоторые могут оказаться "лишними". Эти "лишние" ЛЭ часто можно использовать вместо ввода новых корпусов.

Здесь многое зависит от опыта и квалификации, но ряд стандартных приемов все-таки можно выделить:

* использование схем И-НЕ, ИЛИ-НЕ, НЕ путем объединения входов,
* использование XOR/XNOR, как НЕ, путем подачи "1"/"0" на один вход,
* двойная инверсия для замены И/ИЛИ на И-НЕ/ИЛИ-НЕ,
* замена ИЛИ на И-НЕ с последующей инверсией,
* замена И на ИЛИ-НЕ с последующей инверсией,
* замена ИЛИ-НЕ на И-НЕ – только, если при этом *не увеличивается* количество корпусов и в схеме остается *один тип И-НЕ*.

Каждый из этих приемов или даже несколько из них могут встретиться в данной Л.Р.

***2.3.6.*** *Задание на самостоятельную подготовку.*

В качестве допуска к работе представляется ***бланк*** ДОПУСК/РЕЗУЛЬТАТ со всеми указанными позициями, выполненный на А4. Образец в ПРИЛОЖЕНИИ 6.1.

1) Все комбинации булевых переменных в ТИ в порядке возрастания двоичного кода должны быть проставлены *заранее*.

2) В карте Карно должна быть разметка: двоичных кодов по краям и десятичных кодов – маленькими числами в углах ячеек.

***!!!*** Бланк допуска одновременно является ***чистовым*** бланком предъявления результатов работы. По окончанию Л.Р. он сдается преподавателю и является *единственным* свидетельством выполнения работы.

Бланки с исправлениями и пояснениями "что там написано" НЕ ПРИНИМАЮТСЯ.

Образец заполнения, там же в ПРИЛОЖЕНИИ 6.1, как пример выполнения работы.

**3. ЗАДАНИЕ НА РАБОТУ В ЛАБОРАТОРИИ**.

***3.1.*** *Получение ТИ цифрового устройства.*

***3.1.1.*** *Описание работы со схемой.*

Открыть файл Lab#06varX.ewb (Рис.6.4).

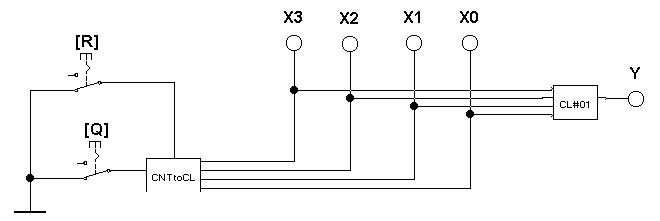


Рис.6.4. Схема для получения ТИ ЦУ.

При *любом* нажатии клавиши **Q** на вход модуля CNTtoCL подается тактовый импульс, который увеличивает состояние внутреннего счетчика на 1.

Состояние выходов Х3 ÷ Х0 отражает циклический двоичный счет от 0 – 0 – 0 – 0 до 1 – 1 – 1 – 1. Затем счетчик сбрасывается в нуль и счет начинается сначала. Клавишей **R** можно произвести сброс в 0 из любого состояния.

Выходы счетчика Х3 ÷ Х0 – это входы ЦУ. Выход ЦУ – на индикаторе Y.

***3.1.2.*** *Заполнение ТИ и карты Карно.*

После очевидных действий по заполнению ТИ перенести результаты в карту Карно.

***3.1.3.*** *Минимизация в карте Карно.*

Для упрощения проверки и нахождения ошибок минимизации составить таблицы, характеризующие этот процесс:

В ячейке "Состав блока" перечисляются по *десятичным кодам* все входящие в него элементы.

В ячейке "Результат минимизации" описывается по *кодам входных переменных* ***один*** элемент, который получился из этого блока.

Сумма содержимого *всех* ячеек "Результат минимизации" и даст требуемую БФ.

***3.1.4.*** *Пример последовательности действий.*

При анализе схемы модуля получилась ТИ, в которой "1" соответствуют строкам (комбинациям Х3 ÷ Х0) с десятичными кодами: 0, 1, 4, 5, 7, 8, 9, 12, 13, 15

1) Сокращенная запись: Σ 0, 1, 4, 5, 7, 8, 9, 12, 13, 15

2) Исходная карта Карно – Рис.6.5(а).



Рис.6.4. Карты Карно: (а) – исходная, (b) – с выделенными блоками.

3) Выделение блоков для минимизации

Блоки-8 в карте Карно.

|  |  |
| --- | --- |
| Состав блока | Результат минимизации |
| 0 – 1 – 4 – 5 – 8 – 9 – 12 – 13 |  |

Блоки-4 в карте Карно.

|  |  |
| --- | --- |
| Состав блока | Результат минимизации |
| 5 – 7 – 13 – 15 |  |

4) Минимизированная БФ



***!!!*** Чистовой бланк ДОПУСК/РЕЗУЛЬТАТЫ должен содержать только исходную карту Карно – Рис.6.4b (можно даже без выделения блоков) и минимизированную БФ.

Промежуточные действия – выделение и описание блоков для минимизации – должны быть в отчете.

***3.1.5.*** *Синтез ЦУ по минимизированной БФ.*

Все варианты разные, но имеют абсолютно равный уровень сложности. Если использовать все возможности минимизации, то *у каждого* должна получиться схема из двух корпусов с вариантами:

* 1 корпус XOR/XNOR + 1 корпус И-НЕ/ИЛИ-НЕ,
* 1 корпус И-НЕ + 1 корпус ИЛИ-НЕ,
* 2 корпуса И-НЕ ( пять ЛЭ типа И-НЕ – это уже два корпуса!).

Возможно (или очевидно) не все смогут получить этот результат, но следующий пункт должны выполнить все. Поэтому выполняйте пп.3.1.6 на достигнутом уровне с соответствующей отметкой в бланке ДОПУСК/РЕЗУЛЬТАТ.

***3.1.6.*** *Верификация ЦУ.*

Схема на основе окончательного вида БФ (на любом уровне минимизации) должна быть нарисована с использованием палитры *Logic Gates*. После этого проводится проверка одинаковых значений Y при одинаковых комбинациях (X3 ÷ X0) для созданной и контрольной схемы (которая в модуле). После того, как схема нарисована:

* отключить индикатор Y от выхода модуля,
* подключить входы ЦУ к переменным X3 ÷ X0,
* переменные X3 ÷ X0 одновременно подаются на модуль и ЦУ,
* выходы модуля и ЦУ подать на входы схемы XOR или XNOR (по выбору),
* индикатор Y подключить к выходу выбранной схемы XOR или XNOR.

Объяснить в отчете ожидаемый результат на выходе в зависимости от выбора XOR или XNOR. Включить схему, проверить результат верификации и предъявить работающую схему преподавателю.

В отчете должна быть схема со стандартными УГО – Рис.6.2