

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ “БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИНФОРМАТИКИ И ЭЛЕКТРОНИКИ”

Кафедра защиты информации

Контрольная работа
по дисциплине

ЦИФРОВЫЕ И МИКРОПРОЦЕССОРНЫЕ УСТРОЙСТВА

Выполнил

Проверил

г. Минск 2007

ОГЛАВЛЕНИЕ

	Введение.....	3
1	Логические основы синтеза цифровых устройств.....	4
1.1	Основные понятия и определения алгебры-логики.....	4
1.2	Минимизация логических функций.....	9
2	Разработка преобразователя кода.....	12
2.1	Вычисление аналитической модели преобразователя.....	12
2.2	Минимизация аналитической модели.....	13
3	Синтез преобразователя кода в булевом базисе.....	17
3.1	Обзор элементной базы.....	17
3.2	Синтез схемы по выбранному критерию.....	18
3.3	Оценка разработки по энергопотреблению быстродействию и размерам.....	21
	Заключение.....	23
	Литература.....	24
	Приложения	
	1. Схема электрическая принципиальная	
	2. Перечень элементов	

ВВЕДЕНИЕ

Элементную базу цифровых устройств (ЦУ) составляют интегральные схемы (ИС). Со времени их изобретения (1956 год, США) ИС постоянно совершенствуются и усложняются. Характеристикой сложности ИС является уровень интеграции, оцениваемый либо числом базовых логических элементов, либо числом транзисторов, которые могут быть реализованы в кристалле.

С ростом уровня интеграции ИС в проектировании всё больше усиливается аспект, который можно назвать интерфейсным проектированием. Задачей разработки ИС становится составление блоков из субблоков стандартного вида путём правильного их соединения. Успешное проектирование требует хорошего знания номенклатуры и параметров элементов, узлов и устройств цифровой аппаратуры и привлечения автоматизированного проектирования (САПР) для создания сложных схем.

ИС широкого применения изготавливаются по технологии КМОП, ТТЛШ, ЭСЛ и др.

Элементы КМОП обладают рядом уникальных параметров (малая потребляемая мощность, высокая помехоустойчивость, широкие допуски на величину питающих напряжений, высокое быстродействие при небольших ёмкостных нагрузках). Эти элементы доминируют в схемах внутренних областей БИС/СБИС.

Элементы ТТЛШ в основном применяются в области периферийных устройств, где требуется передача сигналов по внешним цепям, испытывающим значительную ёмкостную нагрузку.

Элементы ЭСЛ обеспечивают максимальное быстродействие, но ценой повышения потребляемой мощности, что снижает достижимый уровень интеграции.

В данной контрольной работе будут рассмотрены основы синтеза цифровых комбинационных схем.

1. ЛОГИЧЕСКИЕ ОСНОВЫ СИНТЕЗА ЦИФРОВЫХ УСТРОЙСТВ

1.1 ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ АЛГЕБРЫ ЛОГИКИ

Основные логические операции. В цифровых устройствах широко применяются элементы, которые выполняют определенные логические операции. Такие элементы называются логическими (ЛЭ).

Для описания логических операций используется математический аппарат, получивший название алгебры логики или булевой алгебры (в честь ее разработчика — ирландского математика Джорджа Буля). Алгебра логики изучает взаимосвязь между простыми высказываниями, образующими сложные высказывания. С точки зрения алгебры логики простое высказывание может иметь только два значения — истинное или ложное. Одно из этих значений принимается за 1, второе — за 0. Следовательно, простое высказывание является двоичной переменной.

Основными логическими операциями являются:

Операция логического отрицания НЕ (инверсия) преобразует истинное высказывание в ложное или наоборот. Символически операция логического отрицания обозначается в виде черточки над аргументом: $y = \bar{x}$. Такое выражение читается: «у равен НЕ x».

Операция логического отрицания имеет два исхода в зависимости от значения аргумента: а) если $x=0$, то $y=1$ и б) если $x=1$, то $y=0$.

Операция логического сложения ИЛИ (дизъюнкция) используется для образования сложного высказывания из простых. При этом сложное высказывание будет истинным, если истинно хотя бы одно из простых высказываний, и ложным, если ложны все простые высказывания. Символически операция ИЛИ обозначается выражениями:

$$y = x_1 + x_2 + x_3 + \dots \text{ или } y = x_1 \vee x_2 \vee x_3 \vee \dots$$

Читается: «у равен x_1 или x_2 или $x_3 \dots$ »

Результаты логической операции ИЛИ для всех возможных комбинаций двух аргументов приведены в табл. 1.1.

Операция логического умножения И (конъюнкция) тоже используется для образования сложного высказывания из простых, но при этом сложное высказывание считается истинным тогда и только тогда, когда истинны все простые высказывания.

Символически операция И обозначается выражениями:

$$y = x_1 \cdot x_2 \cdot x_3 \cdot \dots \text{ или } y = x_1 \wedge x_2 \wedge x_3 \wedge \dots$$

Читается: «у равен x_1 и x_2 и $x_3 \dots$ »

Результаты логической операции И для двух аргументов также приведены в табл. 1.1.

Операция отрицания логического сложения ИЛИ — НЕ, называемая также «стрелкой Пирса», образует сложное высказывание из простых в соответствии со следующим правилом (табл. 1.1): сложное высказывание истинно лишь в том случае, когда ложны все образующие его простые высказывания, и ложно, если истинно хотя бы одно из простых высказываний.

Символически операция ИЛИ — НЕ обозначается выражениями:

$$y = \overline{x_1 + x_2} \text{ или } y = x_1 \downarrow x_2 \text{ или } y = \overline{x_1 \vee x_2}$$

Операция отрицания логического умножения И — НЕ, известная также под названием «штрих Шеффера», образует сложное высказывание из простых согласно правилу: сложное высказывание истинно, если ложно хотя бы одно из простых высказываний, и ложно, если все простые высказывания истинны (см. табл. 5.1).

Символически операция И — НЕ обозначается выражениями:

$$y = \overline{x_1 \cdot x_2} \text{ или } y = x_1 / x_2 \text{ или } y = \overline{x_1 \wedge x_2}$$

Операция ЗАПРЕТ представляет сложное высказывание, которое истинно только тогда, когда первое из двух высказываний истинно, а второе — ложно. Символически операция ЗАПРЕТ обозначается выражениями:

$$y = x_1 \cdot \overline{x_2} \text{ или } y = x_1 \wedge \overline{x_2}$$

Результаты логической операции ЗАПРЕТ приведены в табл. 1.1.

Таблица 1.1 Результаты логических операций ИЛИ, И, ИЛИ-НЕ, И-НЕ и ЗАПРЕТ

X ₁	X ₂	Y				
		ИЛИ	И	ИЛИ-НЕ	И-НЕ	ЗАПРЕТ
0	0	0	0	1	1	0
0	1	1	0	0	1	0
1	0	1	0	0	1	1
1	1	1	1	0	0	0

Описание работы цифровых устройств. Для описания алгоритмов работы цифровых устройств применяется соответствующий математический аппарат — булева алгебра или алгебра логики.

Рассмотрим некоторое устройство, на входе которого присутствует n -разрядный двоичный код $Z_{n-1}Z_{n-2} \dots Z_1Z_0$, а на выходе соответственно m -разрядный двоичный код $w_{m-1}w_{m-2} \dots w_1w_0$. Для того чтобы описать поведение этой схемы, необходимо определить зависимость каждой из m выходных переменных w_i от входного двоичного кода $Z_{n-1}Z_{n-2} \dots Z_1Z_0$.

Зависимость выходных переменных w_i , выраженная через совокупность входных переменных $Z_{n-1}Z_{n-2} \dots Z_1Z_0$ с помощью операций алгебры логики, носит название *функции алгебры логики* (ФАЛ).

Существуют следующие способы записи ФАЛ: в виде таблицы истинности, в виде алгебраического выражения.

Таблица истинности содержит все возможные комбинации входных переменных $Z_3Z_2Z_1Z_0$ и соответствующие им значения выходных переменных w_i . При описании ФАЛ алгебраическим выражением используются две стандартные формы её представления. *Дизъюнктивной нормальной формой* (ДНФ) называется логическая сумма элементарных логических произведений, в

каждое из которых аргумент или его инверсия входит один раз. Получена ДНФ может быть из таблицы истинности с использованием следующего алгоритма:

а) для каждого набора переменных, на котором ФАЛ равна единице, записывают элементарные логические произведения входных переменных, причём переменные, равные нулю, записывают с инверсией. Полученные произведения называют конституентами единицы;

б) логически суммируют все конституенты единицы.

Дизъюнктивную нормальную форму, полученную суммированием конституент единицы, называют совершенной (СДНФ).

Конъюнктивной нормальной формой (КНФ) называется логическое произведение элементарных логических сумм, в каждое из которой аргумент или его инверсия входит один раз. Получена КНФ может быть из таблицы истинности с использованием следующего алгоритма:

а) для каждого набора переменных, на котором ФАЛ равна нулю, записывают элементарные логические суммы входных переменных, причём переменные, равные единице, записывают с инверсией. Полученные суммы называют конституентами нуля;

б) логически перемножают все полученные конституенты нуля.

Конъюнктивную нормальную форму, полученную перемножением конституент нуля, называют совершенной (СКНФ).

Записав ФАЛ в виде алгебраического выражения для выхода устройства, можно наиболее быстро реализовать логическую схему устройства. Но при этом устройство будет обладать избыточностью элементов и иметь невысокие технические характеристики (потребляемая мощность, быстродействие). Поэтому исходные ФАЛ необходимо минимизировать.

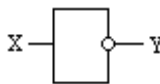
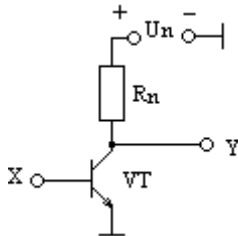
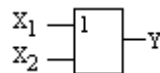
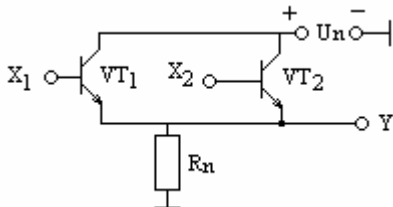
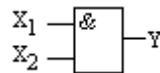
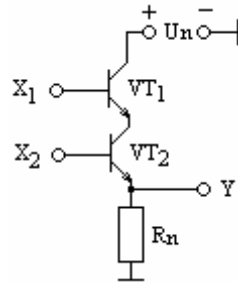
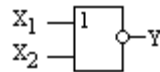
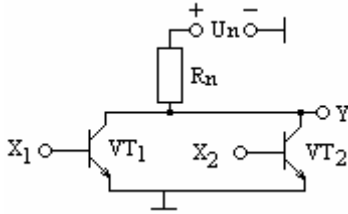
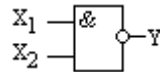
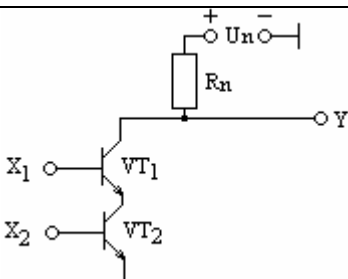
Основные логические элементы. Логические элементы работают с двоичным кодированием информации, которое характеризуется двумя уровнями напряжения двоичной переменной, обозначаемыми цифрами 1 (высокий) и 0 (низкий уровень). В зависимости от уровня напряжения, при котором воспринимается или вырабатывается информация, различают прямые и инверсные входы и выходы ЛЭ. *Прямым* считается такой вход (или выход), на котором двоичная переменная имеет значение 1, когда уровень напряжения на этом входе (выходе) соответствует состоянию, принятому за 1. Если двоичная переменная на входе (выходе) имеет значение 1 при уровне напряжения на нем, соответствующем состоянию, принятому за 0, то такой вход (выход) называется *инверсным*.

Логические элементы обычно выполняются на ИМС, в которых используется положительная (позитивная) логика, когда логической единице соответствует высокий потенциальный уровень, а логическому нулю — низкий потенциальный уровень. Если логической единице соответствует низкий потенциальный уровень, а логическому нулю — высокий, такую логику называют отрицательной, или негативной. Очевидно, что входы и выходы,

являющиеся прямыми в положительной логике, будут инверсными в отрицательной логике, и наоборот.

На принципиальных схемах ЛЭ изображаются в виде прямоугольника (таблица 1.2). В верхней части прямоугольника указывается символ функции: 1 — для логических функций НЕ и ИЛИ и & — для логической функции И. Входы и выходы изображаются линиями, проведенными перпендикулярно к боковым сторонам прямоугольника (входы — с левой стороны, выходы — с правой). Если вход или выход является инверсным, то в месте пересечения изображающей его линии со стороной прямоугольника ставится кружок. В таблице 1.2. приведены буквенные и графические обозначения основных логических элементов, их таблицы истинности и примеры технической реализации.

Таблица 1.2 Простейшие логические элементы.

Тип операций	Обозначение элемента		Таблица истинности	Пример технической реализации															
	буквенное	графическое																	
НЕ	ЛН		<table><tr><th>X</th><th>Y</th></tr><tr><td>1</td><td>0</td></tr><tr><td>0</td><td>1</td></tr></table>	X	Y	1	0	0	1										
X	Y																		
1	0																		
0	1																		
ИЛИ	ЛЛ		<table><tr><th>X1</th><th>X2</th><th>Y</th></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	X1	X2	Y	1	0	1	0	1	1	1	1	1	0	0	0	
X1	X2	Y																	
1	0	1																	
0	1	1																	
1	1	1																	
0	0	0																	
И	И		<table><tr><th>X1</th><th>X2</th><th>Y</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	X1	X2	Y	1	1	1	1	0	0	0	1	0	0	0	0	
X1	X2	Y																	
1	1	1																	
1	0	0																	
0	1	0																	
0	0	0																	
ИЛИ-НЕ	ЛЕ		<table><tr><th>X1</th><th>X2</th><th>Y</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table>	X1	X2	Y	1	1	0	1	0	0	0	1	0	0	0	1	
X1	X2	Y																	
1	1	0																	
1	0	0																	
0	1	0																	
0	0	1																	
И-НЕ	ЛА		<table><tr><th>X1</th><th>X2</th><th>Y</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table>	X1	X2	Y	1	1	0	1	0	1	0	1	1	0	0	1	
X1	X2	Y																	
1	1	0																	
1	0	1																	
0	1	1																	
0	0	1																	

На входах и выходах могут действовать импульсные или постоянные напряжения, соответствующие логическим значениям 0 и 1. В связи с этим различают *импульсные и потенциальные ЛЭ*.

При проектировании схем на дискретных компонентах в качестве базовых элементов являлись логические схемы И, ИЛИ, НЕ на различное число входов (рис. 1.1).

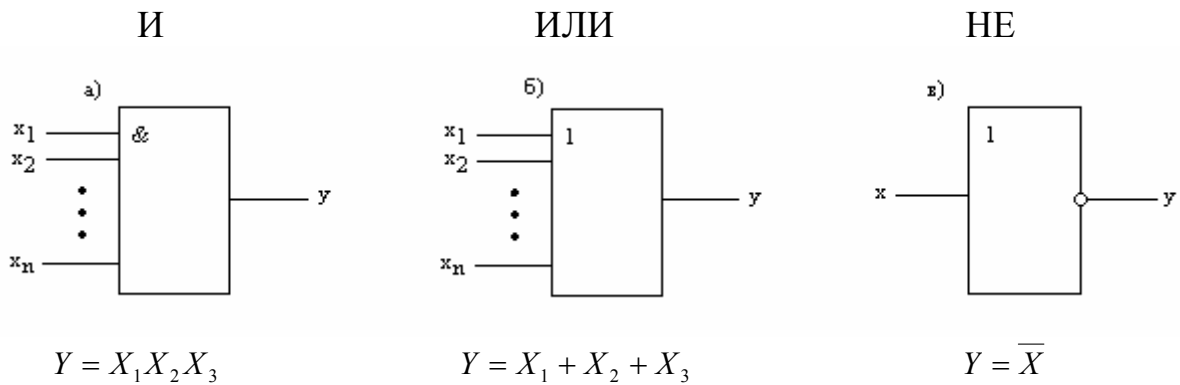


Рис. 1.1. Условные обозначения логических элементов И (а); ИЛИ (б); НЕ (в)

Функциональной полнотой обладают и другие наборы логических функций. Например, $\overline{X_1 \cdot X_2 \cdot \dots \cdot X_n}$ - функция И-НЕ, носящая название функции Шеффера, или $\overline{X_1 + X_2 + \dots + X_n}$ - функция ИЛИ-НЕ, называемая также функцией Пирса (рис. 1.2).

Для удобства построения сложных логических схем имеются базовые элементы, выполняющие и более сложные логические функции.

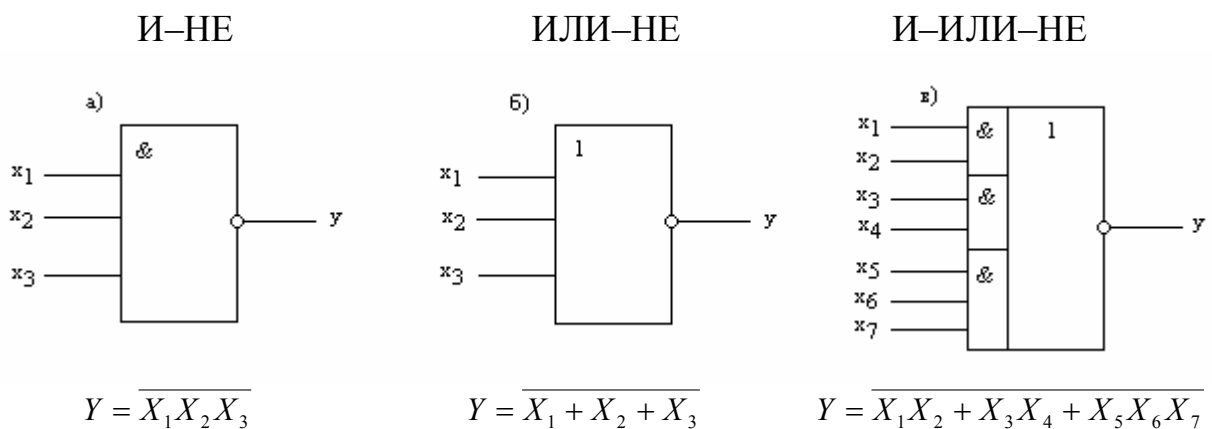


Рис. 1.2. Условные обозначения логических элементов И-НЕ (а); ИЛИ-НЕ (б); И-ИЛИ-НЕ (в)

Например, логические схемы расширенной логики реализуют функцию типа $\overline{X_1 X_2 + X_3 X_4 + X_5 X_6 X_7}$.

1.2 МИНИМИЗАЦИЯ ЛОГИЧЕСКИХ ФУНКЦИЙ

Целью минимизации логической функции является уменьшение стоимости ее технической реализации. Следует отметить, что сам критерий, в соответствии с которым выполняется минимизация ФАЛ, далеко не однозначен и зависит как от типа решаемой задачи, так и уровня развития технологии. Так, в те времена, когда цифровые устройства строились на дискретных элементах, минимизация числа этих элементов и числа построенных на их основе элементарных логических узлов однозначно определяла и уменьшение стоимости технической реализации. С появлением БИС и СБИС, стоимость которых определяется в основном площадью схемы на кристалле и мало зависит от числа входящих в нее транзисторов и других элементов, критерии минимизации ФАЛ претерпели существенные изменения. На первое место при проектировании самих ИС выдвигается требование регулярности их внутренней структуры и минимизация числа внешних соединений даже за счет увеличения числа элементов и внутренних соединений. Эти требования диктуются требованиями повышения надежности электронных средств.

Однако при проектировании аппаратуры с применением БИС и СБИС требование уменьшения числа корпусов ИС и их межсоединений по-прежнему остается весьма важным.

Требование уменьшения числа элементарных ЛЭ, входящих в разрабатываемое устройство, в настоящее время также не потеряло своей актуальности. Объясняется это все более широким использованием при проектировании электронных средств программируемых логических СБИС широкого применения и полужаказных СБИС на основе базовых матричных кристаллов. Эти СБИС и БИС, как правило, содержат отдельные некоммутированные между собой элементарные ЛЭ, например 2И—НЕ или 2ИЛИ—НЕ, или просто наборы транзисторов, резисторов и диодов, которые могут быть соединены между собой в соответствии с заданным алгоритмом обработки логических сигналов. Поскольку число элементов в одной СБИС задано из технологических соображений, то минимизация ФАЛ по критерию уменьшения числа используемых элементов позволяет на одном кристалле решать более сложные задачи логической обработки сигналов, т. е. в конечном счете уменьшать число требуемых ИС и связей между ними. Это снижает стоимость и повышает надежность электронной аппаратуры.

Рассмотрим метод минимизации ФАЛ с использованием карт Вейча-Карно, позволяющий провести минимизацию ФАЛ по критерию уменьшения числа элементарных ЛЭ.

Данный метод базируется на табличном виде представления ФАЛ. Он широко используется при ручной, без применения ЭВМ, минимизации ФАЛ, число переменных в которой обычно не превышает пяти.

Карта Карно — это прямоугольная таблица, число клеток в которой для ФАЛ n -переменных равно 2^n , каждой из клеток поставлен в соответствие некоторый набор входных переменных, причем рядом расположенным клеткам соответствуют соседние наборы входных переменных (кодов), а в самих клетках записаны значения функции, определенные для этих кодов.

Рассмотрим построение карт Карно для функций трех переменных (рис. 2.1). Она содержит восемь клеток. Наборы входных переменных, соответствующие крайним левому и правому столбцам, являются соседними. Поэтому данную карту удобно представить как поверхность цилиндра и она, в отличие от карты двух переменных, является объемной фигурой.

	x_1		\overline{x}_1	
x_0	$f(\overline{x}_2, x_1, x_0)$	$f(x_2, x_1, x_0)$	$f(\overline{x}_2, \overline{x}_1, x_0)$	$f(x_2, \overline{x}_1, x_0)$
\overline{x}_0	$f(\overline{x}_2, x_1, \overline{x}_0)$	$f(x_2, x_1, \overline{x}_0)$	$f(\overline{x}_2, \overline{x}_1, \overline{x}_0)$	$f(x_2, \overline{x}_1, \overline{x}_0)$
	\overline{x}_2	x_2	\overline{x}_2	x_2

Рис. 2.1. Карта Карно для функций трех переменных

Минимизация полностью определенной ФАЛ. При минимизации ФАЛ используют либо её нулевые, либо единичные значения. В обоих случаях получают равносильные выражения, которые, однако, могут отличаться по числу членов (т. е. цене) и выполняемым логическим операциям.

Алгоритм минимизации ФАЛ сводится к следующему.

1. На карте Карно ФАЛ n -переменных выделяют прямоугольные области, объединяющие выбранные значения функции (лог. 0 или лог. 1). Каждая область должна содержать 2^k клеток, где k — целое число. Выделенные области могут пересекаться, т. е. одна или несколько клеток могут включаться в различные области.

2. Каждая из выделенных областей представляется самостоятельным логическим произведением переменных, значения которых в рамках выделенной области остаются постоянными. Каждое произведение содержит $n-k$ переменных и носит название *импликанты*.

3. Из полученного множества выбирают минимальное число максимально больших областей, включающих все выбранные значения ФАЛ.

4. Логически суммируют импликанты, соответствующие выбранным областям. Полученная сумма образует МДНФ, т. е. является покрытием ФАЛ минимальной стоимости (покрытием Квайна).

При объединении клеток с единичными значениями ФАЛ получают МДНФ самой функции, а при объединении клеток с нулевыми значениями ФАЛ — МДНФ функции, инверсной заданной. Последнее легко объясняется при помощи одной из приведенных ранее теорем алгебры логики, согласно которой $x + \overline{x} = 1$.

Очевидно, что если полностью определенная ФАЛ n -переменных принимает значение 1 на m наборах переменных, то на остальных $2^n - m$ входных наборах ее значение равно нулю. Следовательно, объединение 0 значений согласно правилам записи ДНФ приведет к получению функции, инверсной заданной.

Применяя к полученной инверсной минимальной форме теоремы Де-Моргана $\overline{X_1 \vee X_2} = \overline{X_1} \cdot \overline{X_2}$, получаем минимальную функцию, записанную в виде КНФ.

Минимизация не полностью определенной ФАЛ. В некоторых случаях неполнота определения функций (или одной функции) представляет дополни-

тельные возможности для их минимизации. Базируются эти возможности на следующих соображениях. Поскольку избыточные наборы никогда не появятся на входе устройства, то и соответствующие им значения функций также никогда не появятся на выходе этого устройства. Следовательно, в таблице истинности избыточным набором можно условно поставить в соответствие любое значение истинности функции (0 или 1), поскольку это никак не повлияет на работу устройства. Говорят, что в этом случае функции *доопределяются*. Возникает вполне естественная мысль доопределять функции не бессистемно, а целенаправленно, так, чтобы формула доопределенной функции упростилась в максимальной степени с помощью стандартных минимизационных процедур. При этом можно рекомендовать следующий порядок действий для каждой из функций:

- 1) произвести запись значений функций (как единиц, так и нулей) для всех избыточных наборов аргументов в карту Карно;
- 2) доопределить функцию по диаграмме либо единицами, либо нулями так, чтобы отмеченные клетки составляли по возможности наиболее крупные совокупности прямоугольной конфигурации по 2^k -клеток. Необходимо следить, чтобы в отмеченные конфигурации обязательно входила хотя бы одна реализация функции, соответствующая избыточному набору (в противном случае в функции появятся лишние члены);
- 3) по карте Карно производится запись минимальной ДНФ (или КНФ) доопределенной функции обычными способами.

2. РАЗРАБОТКА ПРЕОБРАЗОВАТЕЛЯ КОДА

2.1 ВЫЧИСЛЕНИЕ АНАЛИТИЧЕСКОЙ МОДЕЛИ

ПРЕОБРАЗОВАТЕЛЯ

По заданию устройство имеет четырёхразрядный информационный вход и шестиразрядный информационный выход и предназначено для преобразования кода Ваттса в стартстопный распределительный код. ФАЛ представлены в виде последовательностей двоичных чисел.

Таблица истинности содержит все возможные комбинации входных переменных $x_3x_2x_1x_0$ и соответствующие им значения выходных переменных y_i , называется таблицей истинности или комбинационной таблицей. Предварительно составим таблицу истинности для ФАЛ проектируемого устройства (таблица 2.1).

Таблица 2.1. Таблица истинности устройства

N	x_4	x_3	x_2	x_1	y_6	y_5	y_4	y_3	y_2	y_1
1	0	0	0	0	1	1	0	0	0	0
2	0	0	0	1	1	0	0	1	1	0
3	0	0	1	1	1	0	1	0	1	0
4	0	0	1	0	1	0	1	1	0	0
5	0	1	1	0	1	1	0	0	1	0
6	1	1	1	0	1	1	0	1	0	0
7	1	0	1	0	1	1	1	0	0	0
8	1	0	1	1	1	0	0	0	1	0
9	1	0	0	1	1	0	0	1	0	0
10	1	0	0	0	1	0	1	0	0	0

Логическое устройство, описываемое таблицей 2.1, имеет четыре выхода. Поведение этого устройства описывается системой из шести ФАЛ. В соответствии с изложенным в п. 1.1, СДНФ имеют вид:

$$y_{6\text{СДНФ}}(x_i) = 1;$$

$$y_{5\text{СДНФ}}(x_i) = \overline{x_4}x_3x_2x_1 \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2x_1;$$

$$y_{4\text{СДНФ}}(x_i) = \overline{x_4}x_3x_2x_1 \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2x_1;$$

$$y_{3\text{СДНФ}}(x_i) = \overline{x_4}x_3x_2x_1 \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2x_1;$$

$$y_{2\text{СДНФ}}(x_i) = \overline{x_4}x_3x_2x_1 \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2\overline{x_1} \vee \overline{x_4}x_3x_2x_1;$$

$$y_{1\text{СДНФ}}(x_i) = 0.$$

(1)

2.2 МИНИМИЗАЦИЯ АНАЛИТИЧЕСКОЙ МОДЕЛИ

Минимизацию структуры такого устройства можем выполнить с использованием карт Карно при отдельной минимизации каждой ФАЛ. Однако с точки зрения всего устройства такая структура, как правило, не будет оптимальной.

С точки зрения минимизации всей структуры необходимо, чтобы цепь формирования каждого выходного сигнала y_j была выполнена не минимальным, а некоторым оптимальным способом, обеспечивающим в конечном счёте минимальность общей структуры устройства. Минимизация в этом случае обеспечивается за счёт использования общих цепей формирования сигнала для получения нескольких выходных функций. Последнее достигается выделением на картах Карно различных выходных функций одинаковых областей.

Составим карты Карно функции четырёх переменных ($x_3x_2x_1x_0$) для шести выходов y_6, y_5, y_4, y_3, y_2 и y_1 (рисунок 2.1). ФАЛ, представленная в виде таблицы 2.1, определена не полностью. Доопределим данную функцию на картах Карно нулями и единицами таким образом, чтобы выделяемые прямоугольные области имели максимальный размер.

На картах Карно выделим прямоугольные области, объединяющие единичные значения функции.

Выход y_6

$x_2 \ x_1$		00	01	11	10
$x_4 \ x_3$	00	1	1	1	1
	01				1
	11				1
	10	1	1	1	1

Выход y_5

$x_2 \ x_1$		00	01	11	10
$x_4 \ x_3$	00	1	0	0	0
	01				1
	11				1
	10	0	0	0	1

		Выход y_4			
		$x_2 \ x_1$			
		00	01	11	10
$x_4 \ x_3$	00	0	0	1	1
	01				0
	11				0
	10	1	0	0	1

		Выход y_3			
		$x_2 \ x_1$			
		00	01	11	10
$x_4 \ x_3$	00	0	1	0	1
	01				0
	11				1
	10	0	1	0	0

		Выход y_2			
		$x_2 \ x_1$			
		00	01	11	10
$x_4 \ x_3$	00	0	1	1	0
	01				1
	11				0
	10	0	0	1	0

		Выход y_1			
		$x_2 \ x_1$			
		00	01	11	10
$x_4 \ x_3$	00	0	0	0	0
	01				0
	11				0
	10	0	0	0	0

Рис. 2.1. Карты Карно для выходных функций y_i

На основании рисунка 2.1 запишем ФАЛ для всех выходов.

Логически просуммируем импликанты, соответствующие выбранным областям. Получим

$$\begin{aligned}
y_{6\text{МДНФ}}(x_i) &= 1; \\
y_{5\text{МДНФ}}(x_i) &= x_3 \vee \overline{x_4 x_2 x_1} \vee \overline{x_4 x_2 x_1}; \\
y_{4\text{МДНФ}}(x_i) &= \overline{x_4 x_3 x_2} \vee \overline{x_4 x_3 x_1}; \\
y_{3\text{МДНФ}}(x_i) &= x_4 x_3 \vee \overline{x_2 x_1} \vee \overline{x_4 x_3 x_2 x_1}; \\
y_{2\text{МДНФ}}(x_i) &= \overline{x_4 x_3} \vee \overline{x_4 x_1} \vee \overline{x_2 x_1}; \\
y_{1\text{МДНФ}}(x_i) &= 0.
\end{aligned}$$

Вследствие функциональной полноты функций Шеффера и Пирса, реализующие их элементы И–НЕ или ИЛИ–НЕ могут представлять любую булеву операцию И, ИЛИ, НЕ и таким образом самостоятельно образовать базис, в котором реализуется любая логическая функция. Это целесообразно с двух точек зрения. Во-первых, при проектировании логических схем можно обойтись одним единственным типом вентиля, что позволяет предельно унифицировать этот процесс. Во-вторых, для большинства серий ТТЛ- и КМОП-логик элемент И–НЕ, как и ИЛИ–НЕ, является базисным и предпочтителен во многих отношениях. Вследствие этого реализация логических схем в базисах И–НЕ и ИЛИ–НЕ получила широкое распространение в практике.

Для реализации функций в базисе И–НЕ запишем над ними символ двойной инверсии и одну из общих инверсий преобразуем по формуле Де–Моргана. Тогда

$$\begin{aligned}
y_{6\text{МДНФ}}(x_i) &= 1; \\
y_{5\text{МДНФ}}(x_i) &= x_3 \vee \overline{x_4 x_2 x_1} \vee \overline{x_4 x_2 x_1} = x_3 \vee \overline{x_4 x_2 x_1} \vee \overline{x_4 x_2 x_1} = x_3 \cdot \overline{x_4 x_2 x_1} \cdot \overline{x_4 x_2 x_1}; \\
y_{4\text{МДНФ}}(x_i) &= \overline{x_4 x_3 x_2} \vee \overline{x_4 x_3 x_1} = \overline{x_4 x_3 x_2} \vee \overline{x_4 x_3 x_1} = \overline{x_4 x_3 x_2} \cdot \overline{x_4 x_3 x_1}; \\
y_{3\text{МДНФ}}(x_i) &= x_4 x_3 \vee \overline{x_2 x_1} \vee \overline{x_4 x_3 x_2 x_1} = x_4 x_3 \vee \overline{x_2 x_1} \vee \overline{x_4 x_3 x_2 x_1} = x_4 x_3 \cdot \overline{x_2 x_1} \cdot \overline{x_4 x_3 x_2 x_1}; \\
y_{2\text{МДНФ}}(x_i) &= \overline{x_4 x_3} \vee \overline{x_4 x_1} \vee \overline{x_2 x_1} = \overline{x_4 x_3} \vee \overline{x_4 x_1} \vee \overline{x_2 x_1} = \overline{x_4 x_3} \cdot \overline{x_4 x_1} \cdot \overline{x_2 x_1}; \\
y_{1\text{МДНФ}}(x_i) &= 0.
\end{aligned} \tag{2}$$

По выражениям (2) построим функциональную схему разрабатываемого устройства (рисунок 2.2).

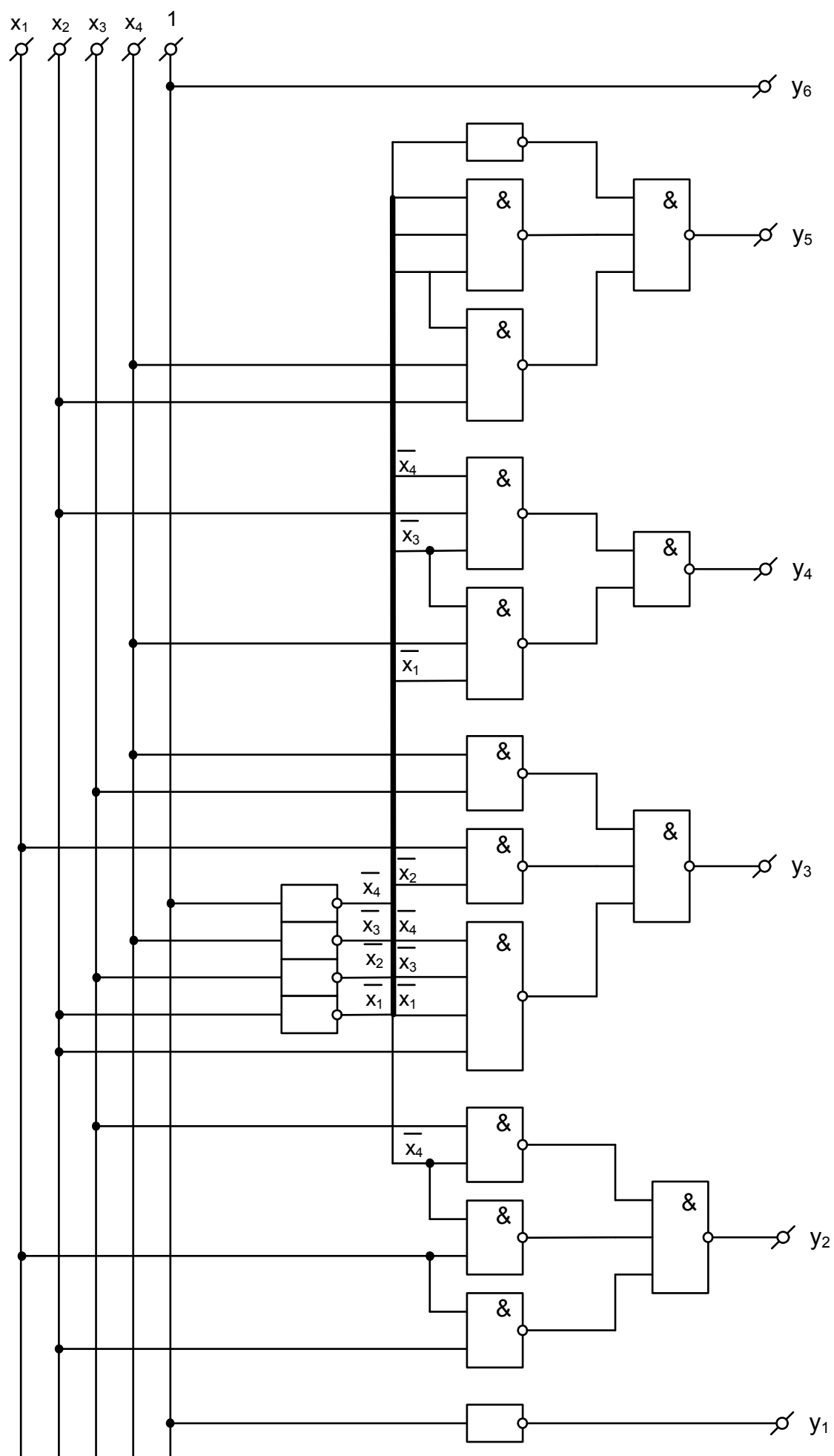


Рис. 2.2. Функциональная схема устройства

3. СИНТЕЗ ПРЕОБРАЗОВАТЕЛЯ КОДА В БУЛЕВОМ БАЗИСЕ

3.1 ОБЗОР ЭЛЕМЕНТНОЙ БАЗЫ

Логические элементы широко представлены в различных типах логики интегральных микросхем (ИМС) – транзисторно-транзисторной (ТТЛ), на комплиментарных полевых транзисторах (КМДП), эмиттерно-связанной (ЭСЛ). Выбор того или иного типа ИМС зависит от назначения, области и условий применения разрабатываемого устройства и определяется оптимизируемым параметром. Наиболее часто в качестве такового рассматривается один из следующих:

- быстродействие;
- потребляемая мощность;
- стоимость;
- надёжность.

Интегральные схемы транзисторно-транзисторной логики в настоящее время являются наиболее распространёнными микросхемами. Длительное время микросхемы ТТЛ составляли элементную базу ЭВМ. Все микросхемы ТТЛ можно разделить на три группы:

- стандартные;
- со структурами Шотки (ТТЛШ);
- усовершенствованные ТТЛШ (более экономичные и быстродействующие).

Широкое распространение ТТЛ ИС обусловлена высоким быстродействием микросхем этого типа при вполне удовлетворительных других эксплуатационных характеристиках.

В качестве основных преимуществ КМДП ИС следует назвать:

- низкую статическую мощность, потребляемую от источника питания;
- высокую работоспособность в широком диапазоне питающих напряжений (3 – 15 В);
- высокую помехозащищённость;
- высокую нагрузочную способность;
- упрощенное сопряжение со слаботочными источниками входного напряжения;
- высокая технологичность изготовления.

При этом КМДП ИС существенно проигрывают микросхемам транзисторно-транзисторной и эмиттерно-связанной логики по быстродействию. Поэтому КМДП ИС применяются там, где энергетический фактор является решающим при выборе элементной базы.

Интегральные микросхемы на основе эмиттерно-связанной логики получили широкое распространение в качестве элементной базы быстродействующей вычислительной и радиоэлектронной аппаратуры. Достоинствами микросхем ЭСЛ являются:

- хорошая схемно-техническая отработанность и, как следствие, сравнительно невысокая стоимость изготовления;

высокое быстродействие при средней потребляемой мощности или сверхвысокое быстродействие при большой потребляемой мощности;
малая энергия переключения;
высокая относительная помехоустойчивость;
высокая стабильность параметров при изменении рабочей температуры и напряжения питания;
большая нагрузочная способность;
независимость тока потребления от частоты переключения;
удобство применения в условиях повышенной плотности компоновки с использованием многослойного печатного монтажа.

Опыт проектирования аппаратуры показывает, что ЭСЛ ИС оптимальны для построения быстродействующих радиоэлектронных устройств и менее эффективны при разработке радиоэлектронных устройств малого и среднего быстродействия.

Исходя из приведенных рассуждений наиболее целесообразным представляется использовать для разработки заданного регистра ИМС транзисторно-транзисторной логики.

3.2 СИНТЕЗ СХЕМЫ ПО ВЫБРАННОМУ КРИТЕРИЮ

Критерием выбора микросхем определим быстродействие. Из справочника [2] видно, что наиболее быстродействующие микросхемы ТТЛ-логики из серии 1533.

На основании выражений (2) рисунка (2.2), для реализации функций возбуждения триггеров необходимо использовать микросхемы, элементы которых выполняют функции 2И-НЕ, 3И-НЕ, 4И-НЕ. Такими микросхемами являются 1533ЛА3, 1533ЛА4, 1533ЛА1, условные графические обозначения и технические характеристики которых приведены ниже.

Условное графическое обозначение и цоколёвка ИМС 1533ЛА4 показаны на рис. 3.1, основные характеристики микросхемы приведены в таблице 3.1. Микросхема содержит три элемента 3И-НЕ.

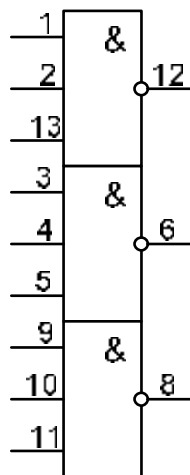


Рис. 3.1 Условное графическое обозначение и цоколёвка ИМС 1533ЛА4

Таблица 3.1 Основные параметры ИМС 1533ЛА4

Напряжение источника питания, В	Основные параметры					
	$U^0_{\text{вых}}, \text{В}$	$U^1_{\text{вых}}, \text{В}$	$t^{1,0}_{\text{зд.р.}}, \text{нс}$	$t^{0,1}_{\text{зд.р.}}, \text{нс}$	$I^1_{\text{пот}}, \text{мА}$	$I^0_{\text{пот}}, \text{мА}$
5	0,4	2,5	14	21	0.6	2.2

Микросхема 1533ЛА3 содержит четыре логических элемента 2И-НЕ. Условное графическое обозначение и цоколёвка ИМС показаны на рис. 3.2, основные характеристики микросхемы приведены в таблице 3.2.

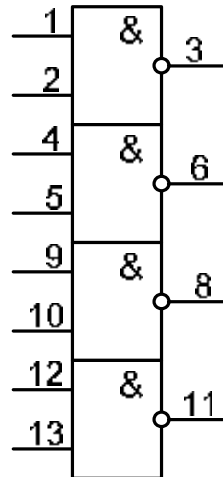


Рис. 3.2. Условное графическое обозначение и цоколёвка ИМС 1533ЛА3

Таблица 3.2. Основные параметры ИМС 1533ЛА3

Напряжение источника питания, В	Основные параметры					
	$U^0_{\text{вых}}, \text{В}$	$U^1_{\text{вых}}, \text{В}$	$t^{1,0}_{\text{зд.р.}}, \text{нс}$	$t^{0,1}_{\text{зд.р.}}, \text{нс}$	$I^1_{\text{пот}}, \text{мА}$	$I^0_{\text{пот}}, \text{мА}$
5	0,4	2,5	14	14	0,85	3,0

Микросхема 1533ЛА1 содержит два логических элемента 4И-НЕ. Условное графическое обозначение и цоколёвка ИМС показаны на рис. 3.3, основные характеристики микросхемы приведены в таблице 3.3.

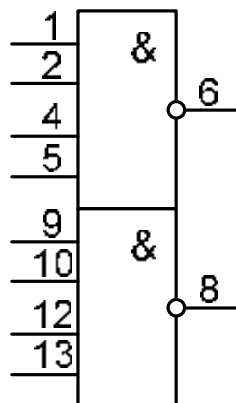


Рис. 3.3. Условное графическое обозначение и цоколёвка ИМС 1533ЛА1

Таблица 3.3. Основные параметры ИМС 1533ЛА1

Напряжение источника питания, В	Основные параметры					
	$U^0_{\text{вых}}, \text{В}$	$U^1_{\text{вых}}, \text{В}$	$t^{1,0}_{\text{зд.р.}}, \text{нс}$	$t^{0,1}_{\text{зд.р.}}, \text{нс}$	$I^1_{\text{пот}}, \text{мА}$	$I^0_{\text{пот}}, \text{мА}$
5	0,4	2,5	28	14	0,4	1,5

Исходя из функциональной схемы на рисунке 2.2, необходимо следующее количество рассмотренных типов ИМС: 1533ЛА1 – 1 шт., 1533ЛА4 – 2 шт., 1533ЛА3 – 2 шт.

Из выбранных ИМС на основании функциональной схемы составим принципиальную схему устройства. Данная схема изображена в приложении 1. На данной схеме применены следующие обозначения

DD1, DD4, DD6 – ИМС 1533ЛА3;

DD2, DD3 – ИМС 1533ЛА4,

DD5 – ИМС 1533ЛА1.

Перечень элементов приведён в приложении 2.

3.3 ОЦЕНКА РАЗРАБОТКИ ПО ЭНЕРГОПОТРЕБЛЕНИЮ, БЫСТРОДЕЙСТВИЮ И РАЗМЕРАМ

Произведем расчет быстродействия разработанного преобразователя. Для этого просуммируем среднее время задержки распространения сигнала во всех узлах устройства, которые работают последовательно.

Из принципиальной схемы видно, что имеется несколько путей прохождения сигналов. Из анализа быстродействия микросхем, максимальную задержку имеет сигнал, проходящий по пути DD1-DD5-DD3 (сигнал y_3).

Суммарная задержка сигнала y_2 равна

$$t_3 = t_{\text{ЛА3}} + t_{\text{ЛА1}} + t_{\text{ЛА4}} = 14 + \frac{14 + 28}{2} = \frac{14 + 21}{2} = 52.5 \text{ (нс)}.$$

Тогда максимальная частота работы устройства определяется выражением

$$F_{\text{max}} = \frac{1}{t_1} = \frac{1}{52.5 \cdot 10^{-9}} \approx 19.05 \text{ (МГц)}.$$

Произведем расчет средней потребляемой мощности преобразователя кода. Для этого необходимо просуммировать среднюю потребляемую мощность для всех микросхем, входящих в устройство. Так как средняя потребляемая мощность микросхемой зависит от величины напряжения питания, то ее значение приблизительно определяется по формуле

$$P_{\text{ср}} = U_{\text{пит}} \cdot I_{0 \text{ ср}}.$$

Значения $U_{\text{пит}}$ и $I_{0 \text{ ср}}$ указаны в технических характеристиках на микросхему. Результаты расчетов для каждого типа микросхем сведем в таблицу 3.4.

Таблица 3.4. Расчет потребляемой мощности устройством

Тип микросхемы (элемента)	$P_{\text{ср}}$ одного корпуса, мВт	Количество корпусов	Суммарная $P_{\text{ср}}$, мВт
1533ЛА1	4.75	1	4.75
1533ЛА3	9.6	3	28.8
1533ЛА4	7.0	2	14.0
Суммарная мощность:			47.55

Произведём оценку размеров разработанного устройства. Будем полагать, что микросхемы располагаются на одной стороне платы (рис. 3.4).

Примем коэффициент площади ИМС равным $K_{\text{п}}=2$. Тогда площадь платы равна

$$S_{\text{п}} = N \cdot S_0 \cdot K_{\text{п}},$$

где $S_0 = 1.8 \text{ см}^2$ – площадь одной микросхемы,
 N – общее количество микросхем (округлённое до большего чётного).

Рассчитаем площадь платы равна

$$S_{\Pi} = N \cdot S_0 \cdot K_{\Pi} = 6 \cdot 1.8 \cdot 2 = 21.6 \text{ (см}^2\text{)}$$

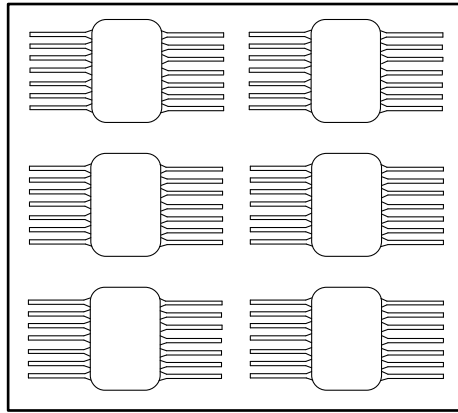


Рис. 3.5. Расположение микросхем на плате

ЗАКЛЮЧЕНИЕ

В результате выполненной контрольной работы было разработано устройство преобразования кода Ваттса в стартстопный распределительный. При синтезе устройства проводилась минимизация с использованием карт Карно.

По функциональной схеме была разработана принципиальная схема устройства.

Элементной базой являются интегральные микросхемы ТТЛШ.

В состав устройства входят 6 микросхем серии 1533.

Устройство преобразования имеет следующие технические характеристики:

напряжение питания – 5 В;

средняя потребляемая мощность – 47.55 мВт;

максимальная частота преобразования – 19.05 МГц,

размер платы – 21.6 см².

ЛИТЕРАТУРА

1. “Цифровая и вычислительная техника”. Под редакцией Э.В.Евреина. Учебник для ВУЗов. М.: Радио и связь, 1991 год.
2. “Аналоговая и цифровая электроника”. Опадчий Ю.Ф. и др. Учебник для ВУЗов. М.: Телеком, 2003 год.
3. Цифровые интегральные микросхемы. Справочник. Богданович М.И и др. Мн.: “Беларусь”, 1991 год.

Приложение 2

Перечень элементов

[illegible]