

ФЕДЕРАЛЬНОЕ АГЕНТСТВО СВЯЗИ  
Северо-Кавказский филиал  
ордена Трудового Красного Знамени федерального государственного бюджетного образовательного учреждения высшего образования  
"Московский технический университет связи и информатики"

---



Методические указания  
к практическим занятиям

## **ПРИМЕНЕНИЕ МИКРОСХЕМ ПАМЯТИ**

Направление подготовки:

09.03.01 Информатика и вычислительная техника

11.03.02 Инфокоммуникационные технологии и системы связи

Профиль:

Многоканальные телекоммуникационные системы

Сети связи и системы коммутации

Защищенные системы и сети связи

Системы радиосвязи и радиодоступа

Вычислительные машины, комплексы, системы и сети

Программное обеспечение и интеллектуальные системы

Ростов-на-Дону  
2019

УДК 681.3.06 (076)  
ББК 32.07

Чикалов А.Н. Применение микросхем памяти. Методические указания к практическим занятиям. Ростов-на-Дону: Северо-Кавказский филиал МТУСИ, 2019.- 35 с.

В пособии изложены методические рекомендации, содержательные материалы и контрольные задания для проведения практических занятий по изучению принципов построения микросхем памяти, блоков памяти, исследованию поведения микросхем, принципов программирования для реализации комбинационных устройств, синтеза динамических схем различного назначения. Приведены основные этапы и примеры разработки устройств на основе ПЗУ. Пособие содержит необходимые справочные материалы.

Методические указания предназначены для студентов, обучающихся по направлениям подготовки 09.03.01 Информатика и вычислительная техника и 11.03.02 Инфокоммуникационные технологии и системы связи, профилей Многоканальные телекоммуникационные системы, Сети связи и системы коммутации, Защищенные системы и сети связи, Системы радиосвязи и радиодоступа, Вычислительные машины, комплексы, системы и сети, Программное обеспечение и интеллектуальные системы.

Пособие предназначено для использования при изучении дисциплин Микропроцессорные системы, Вычислительная техника и информационные технологии, а также может быть использовано преподавателями и студентами при изучении родственных дисциплин и в процессе самостоятельной работы.

Учебное пособие обсуждено и одобрено на заседании кафедры ИВТ  
Протокол №1 от 26.08.2019

Рецензент Зав. кафедрой ИВТ д.т.н. профессор Соколов С.В.

## СОДЕРЖАНИЕ

1. Программирование ПЗУ . . . . .	4
1.1. Построить дешифратор с настраиваемыми выходами . . . . .	7
1.2. Построить дешифратор К555ИД7 на основе ПЗУ . . . . .	10
1.3. Построить дешифратор для семисегментного индикатора . . . .	11
1.4. Построить полный двухразрядный сумматор на основе ПЗУ . .	12
1.5. Построить на ПЗУ неполный дешифратор с прямыми выходами с выборкой по заданным адресам. . . . .	14
2. Использование ПЗУ в динамических устройствах . . . . .	16
2.1. Построить генератор сигнала бедствия, запускающийся по включению питания . . . . .	17
2.2. Создать бегущую строку из четырех цифровых индикаторов . .	18
2.3. Создать бегущую строку с изменяющейся информацией . . . .	21
2.4. Исследовать возможность увеличения длины бегущей строки до 8 индикаторов . . . . .	21
3. Разработка модуля памяти . . . . .	22
3.1. Построение блока дешифрации адресов . . . . .	23
3.2. Построение схем дешифраторов адресов ОЗУ и ПЗУ. . . . .	29
3.3. Разработка блока дешифрации адресов устройств ввода-вывода. . . . .	30
3.4. Построение функциональной схемы модуля памяти . . . . .	32

# 1. ПРОГРАММИРОВАНИЕ ПЗУ

## Цель

1. Выработать практические умения синтеза комбинационных схем с использованием ПЗУ.
2. Совершенствовать практические умения и навыки работы с цифровой техникой, ЭВМ и системами моделирования.
3. Совершенствовать навыки анализа, обобщения и систематизации полученных результатов, навыки составления и оформления отчетных материалов, навыки точного и лаконичного представления докладов на вопросы технического характера.

## Учебные вопросы

- 1.1. Построить дешифратор с настраиваемыми выходами (прямыми либо инверсными).
- 1.2. Построить дешифратор К555ИД7 на основе ПЗУ.
- 1.3. Построить дешифратор для семисегментного индикатора.
- 1.4. Построить полный двухразрядный сумматор на основе ПЗУ.
- 1.5. Построить на ПЗУ неполный дешифратор с прямыми выходами с выборкой по заданным адресам

## Литература для подготовки к занятию

1. Угрюмов Е.П. Цифровая Схемотехника: Учебное пособие для вузов. – СПб.: БХВ-Петербург, 2007. – 800с.
2. Бойко В.И. и др. Схемотехника электронных систем. Цифровые устройства.- СПб.: БХВ-Петербург, 2004.-512с.
3. Потемкин И.С. Функциональные узлы цифровой автоматики. - М.: Энергоатомиздат, 1988.-320с.
4. Схемотехника БИС постоянных запоминающих устройств. /О.А.Петросян, И.Я. Козырь и др.-М.: Радио и связь, 1987.-304с.
5. Полупроводниковые БИС запоминающих устройств: Справочник /Под ред. А.Ю. Гордонова и Ю.А. Дьякова.-М.: Радио и связь, 1987.-360с.
6. Лебедев О.Н. Микросхемы памяти и их применение.- М.: Радио и связь, 1990.-160с.
7. Шило В.Л. Популярныe цифровые микросхемы. – М.: Радио и связь, 1987.-352с.

## Содержание отчета

1. Название работы.
2. Последовательно для каждого из учебных вопросов: название учебного вопроса и перечень тех материалов, которые указаны в конкретном задании;
3. Краткие ответы на те контрольные вопросы для самопроверки, которые ещё не нашли своего отражения в отчете.

### Вопросы для самопроверки

1. Приведите классификацию ИМС полупроводниковой памяти.
2. Как реализованы элементы хранения для масочных, однократно программируемых и перепрограммируемых ПЗУ.
3. Нарисуйте структурную схему ПЗУ. Каково назначение основных структурных элементов? Каково назначение входов и выходов ПЗУ?
4. Назовите основные параметры ПЗУ, охарактеризуйте их физический смысл.
5. Какие типы выходов применяются в микросхемах ПЗУ? Для чего их используют?
6. Какое состояние имеют элементы памяти ПЗУ до программирования? Как это определить?
7. Для каких целей в ИМС памяти создают несколько входов CS? По какой логике они работают?
8. Каково назначение входов и выходов ИМС 155РЕЗ? Каково ее условное графическое обозначение на схемах?
9. Каков порядок считывания данных из ИМС ПЗУ?
10. Для каких целей используют ПЗУ?

### Актуальность занятия

Схемы ПЗУ помимо своего применения по прямому назначению широко используются как элементы для реализации различных сложных схем комбинационной логики. Это устраняет проблему минимизации, сокращает время разработки и удешевляет конструкцию, уменьшает энергопотребление схемы. Поэтому инженеру совершенно необходимо освоить технологию подобных разработок.

### **Микросхема ПЗУ 155РЕЗ**

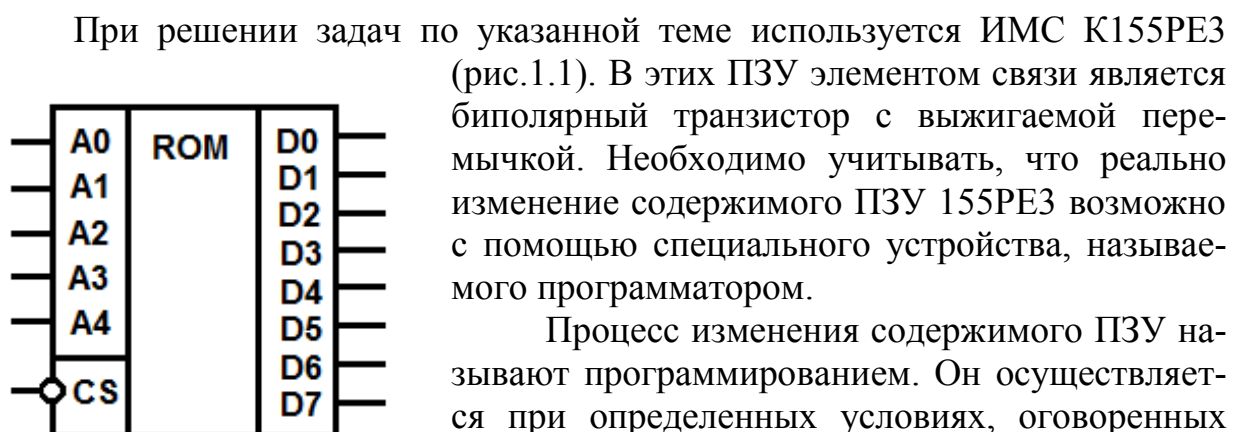


Рис.1.1. Условное графическое обозначение ИМС К155РЕЗ

Процесс изменения содержимого ПЗУ называют программированием. Он осуществляется при определенных условиях, оговоренных технической документацией на соответствующее ПЗУ. При программировании К155РЕЗ в узлах, где должен быть записан 0, через транзистор пропускают импульс тока, достаточный для разрушения перемычки. Такой способ называют электрическим способом программирования. Эта микросхема имеет

Математическая модель микросхемы 155РЕЗ в программе ВАРИАНТ позволяет имитировать процесс программирования и считывания информации из ПЗУ. Для изменения содержимого ПЗУ необходимо вызвать условное графическое изображение ПЗУ на рабочий экран и ввести режим коррекции содержимого ПЗУ. Модель в исходном состоянии содержит 32 нулевых слова (т.е. слова с кодом 00h). Содержимое ПЗУ 155РЕЗ в таком состоянии представлено на рис.1.2.

[illegible]

Рис.1.2. Карта памяти незапрограммированной ИМС К155РЕЗ

### Задание 1.1. Построить дешифратор с настраиваемыми выходами

Настройка дешифратора заключается в выборе типа выходов дешифратора: прямых или инверсных. Выходы должны получаться при подаче настраивающего сигнала на вход устройства. Для этого, естественно, достаточно одного вывода.

Необходимо помнить следующее:

1. У схемы ПЗУ 155РЕЗ число выходов равно восьми. Поэтому при проектировании полного дешифратора количество его выходов не может быть увеличено, а для управления выходами потребуется всего три адресных входа. Они и должны стать адресными входами создаваемого дешифратора. Рационально (для простоты восприятия) для этого выбрать младшие адресные входы ПЗУ: А0, А1, А2. При этом адресовать можно будет только восемь ячеек памяти с адресами 000 – 111 (0h-7h). Нумерация выходов создаваемого дешифратора рационально привязать к номерам выходов схемы памяти. Такая нумерация выходов дешифратора является логически обоснованной;

2. Оставшиеся адресные входы (А3 и А4) своими сигналами способны выделить в адресном пространстве ПЗУ четыре области памяти по восемь ячеек. Эти адресные входы ПЗУ своими значениями как бы разделяют все адресное пространство на четыре части (рис.1.3).

A4	A3	A2	A1	A0	Hex
0	0	0	0	0	00
0	0	0	0	1	01
0	0	0	1	0	02
0	0	0	1	1	03
0	0	1	0	0	04
0	0	1	0	1	05
0	0	1	1	0	06
0	0	1	1	1	07
0	1	0	0	0	08
0	1	0	0	1	09
0	1	0	1	0	0A
0	1	0	1	1	0B
0	1	1	0	0	0C
0	1	1	0	1	0D
0	1	1	1	0	0E
0	1	1	1	1	0F
1	0	0	0	0	10
1	0	0	1	0	11
1	0	1	0	1	12
1	0	1	1	0	13
1	1	0	0	0	14
1	1	0	1	0	15
1	1	1	0	0	16
1	1	1	1	1	17
1	0	0	0	0	18
1	0	0	1	0	19
1	0	1	0	1	1A
1	0	1	1	0	1B
1	1	0	0	1	1C
1	1	1	0	1	1D
1	1	1	1	0	1E
1	1	1	1	1	1F

Рис.1.3. Деление адресного пространства ПЗУ старшими адресными сигналами

Самый старший разряд (А4) своим значением разделяет адресное пространство пополам: при неизменном нулевом сигнале этого разряда могут быть выбраны адреса 00h – 0Fh, а при единичном значении А4 – адреса 10h – 1Fh (см. рис.1.3). Следующий, более младший, адресный вход (А3) аналогично разделяет еще раз пополам каждую уже получившуюся половину адресного пространства: при неизменном нулевом сигнале на А3 и различных сигналах на А4 могут быть выбраны адреса 00h-07h или 10h-17h, а при А3=1, соответственно, 08h-0Fh или 18h-1Fh. Получившиеся при изменениях значений входов А4 и А3 области адресов на рис.1.3 выделены сплошными линиями. В каждой из получившихся четырех областей младшие адресные вхо-

ды могут сформировать абсолютно повторяющиеся множества адресных комбинаций: от 000 до 111.

Следует подчеркнуть, что половинное разделение адресов своим сигналом осуществляет каждый адресный вход. Однако, чем младше разряд, тем более фрагментированные области он выделяет. Только самый старший адресный вход А4 разделяет пространство на две неразрывные части. Более младший адресный вход А3 выделяет каждую половину адресов уже в виде двух частей в разных местах адресного пространства и т.д. Это легко увидеть на рис.1.3;

3. Вход разрешения ПЗУ для создания дешифратора использован быть не может. Это следует из особенностей его воздействия на схему ПЗУ: нет для проектируемого дешифратора такого входа, при единичном сигнале которого все выходы равны бы были единице независимо от любых условий и режимов работы дешифратора. Поэтому на вход CS ПЗУ следует подать постоянный разрешающий сигнал, а логику работы дешифратора моделировать путем программирования содержимого ПЗУ и манипулирования старшими адресными входами;

4. Для проектируемого дешифратора необходим один вход, который будет менять режим работы: дешифратор должен получаться с прямыми либо с инверсными выходами. Для ПЗУ это две области с разными записанными в накопитель данными. Для такого входа можно использовать один из старших адресных входов, например, А3. Он может выбирать либо диапазон адресов 00h-07h и 10h-17h при А3=0, либо 08h-0Fh и 18h-1Fh при А3=1 в зависимости от состояния входа А4. В этих адресных полях должны быть запрограммированы выходы для прямого и инверсного дешифратора соответственно в зависимости от сигнала разрешения, который должен моделироваться входом А4;

5. Оставшийся адресный вход ПЗУ А4 должен быть использован для подачи сигнала разрешения дешифратора. Из имеющихся четырех 8-адресных полей 00h-07h, 10h-17h и 08h-0Fh, 18h-1Fh он может, например, единицей запрещения работы дешифратора выбирать поля 10h-17h и 18h-1Fh. В этих полях следует запрограммировать все пассивные выходы (невыбор) для прямого или инверсного дешифратора соответственно. Для разрешающего же нулевого сигнала будут выбираться поля 00h-07h и 08h-0Fh. В них и следует запрограммировать активные сигналы, соответствующие адресной комбинации на младших адресных входах А0, А1, А3 для своего типа дешифратора.

Задание выполняется по индивидуальным вариантам в соответствии с номером по журналу учебной группы:



Номер по журналу учебной группы	Вход для выбора режима DC	Сигнал для выбора DC с прямыми выходами	Вход для сигнала разрешения DC	Сигнал для разрешения работы DC
1, 16	A3	1	A4	1
2, 18	A4	1	A3	1
3, 17	A3	0	A4	0
4, 14	A4	0	A3	0
5, 12	A3	0	A4	1
6, 11	A4	0	A3	1
7, 20	A3	1	A4	0
8, 13	A4	1	A3	0
9, 15	A3	0	A4	1
10, 19	A4	0	A3	0

Для выполнения задания необходимо:

- выбрать адресный вход ПЗУ для управления выбором режима работы дешифратора и назначить конкретный сигнал для конкретного режима;
- второй адресный вход назначить входом разрешения работы дешифратора и задать тип разрешающего сигнала;
- составить условное графическое обозначение (УГО) дешифратора с указанием входов схемы памяти, которые будут имитировать входы дешифратора;
- составить карту памяти с указанием характера поведения ПЗУ в каждой из четырех областей адресного пространства в соответствии с семантикой работы дешифратора, исходя из комбинации входных сигналов разрешения и режима работы по своему варианту. Карту лучше составлять схематически, выбрав варианты обозначения, например, как показано на рис.1.4. В случае подробной карты может быть утрачена логика разделения адресного пространства;

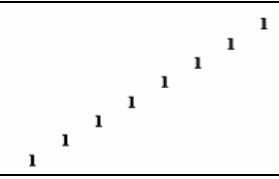
A0	Все единицы, работа запрещена	Прямой DC, работает	0	
A1				
A2				
A3	0	1	0	1
A4	0		1	

Рис.1.4. Варианты схематического обозначения данных для программирования ПЗУ

- запрограммировать каждую из получившихся четырех областей;

- проверить работоспособность схемы на всех режимах работы дешифратора: изменение типа дешифратора, запрет работы дешифратора для каждого типа, дешифрации адреса на всех комбинациях.

В отчете представить:

1. УГО дешифратора с обозначением используемых входов ПЗУ для моделирования его работы;
2. Сигналы для выбора соответствующего режима работы и разрешения работы;
- 3 Схематическую карту программирования ПЗУ.

Вопросы для дополнительного исследования:

1. Можно ли вход CS использовать для реализации разрешения для вновь создаваемого дешифратора? Почему?
2. Что изменится, если A3 и A4 поменять местами для решения задачи выбора режима работы и разрешения работы дешифратора?

### Задание 1.2. Построить дешифратор K555ИД7 на основе ПЗУ

Алгоритм решения задачи:

- выбор (выявление) входов микросхемы памяти, способных обеспечить выполнение функций, аналогичных тем, которые выполняют входы дешифратор K555ИД7;
- программирование соответствующих ячеек памяти ПЗУ;
- проверка функционирования дешифратора на основе ПЗУ.

Для выполнения задания необходимо изменить содержимое ячеек памяти так, чтобы реализовать на микросхеме K155PE3 устройство, выполняющее функции дешифратора 155ИД7 (рис.1.5).

Дешифратор, как известно, это устройство, преобразующее кодовую комбинацию входных сигналов в активный сигнал на одном из выходов (и только на одном!!!).

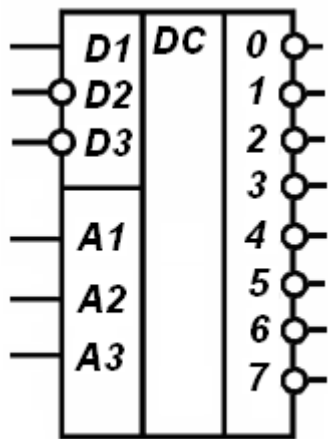


Рис.1.5. ИМС K555ИД7

Активный выходной сигнал дешифратора K155ИД7 - логический ноль. Если одно из слов микросхемы памяти представить в виде 8-разрядного кода, где на одной позиции будет 0, а на остальных 1, то такую кодовую комбинацию вполне можно будет считать комбинацией выходных сигналов дешифратора. Число выходов K155PE3 и K155ИД7 одинаково и равно восьми. Следовательно, для активизации (выбора) одного из выходов дешифратора достаточно трех адресных входов. Как и в задании 1.1. для адресации выхода дешифратора целесообразно выбрать

младшие адресные входы A0, A1, A2. Поэтому в ПЗУ K155PE3, если использовать его для построения дешифратора, остаются незадействованными два старших адресных входа. Их следует использовать для имитации дополнительных входов разрешения.

Остается нереализованным один из входов разрешения дешифратора. Но из свободных входов ПЗУ остается только вход CS. Поэтому следует определить, исходя из логики работы входа CS, какой из входов разрешения дешифратора он способен моделировать. Логике CS в ПЗУ изменить невозможно, она реализована аппаратно, поэтому назначение входов для реализации дешифратора необходимо начинать именно с него.

Оставшиеся входы разрешения дешифратора должны быть реализованы на старших адресных входах ПЗУ A3 и A4 (см. рис.1.1). Им необходимо присвоить смысл конкретного входа разрешения дешифратора из оставшихся двух свободных после использования входа CS. Адресные входы A3 и A4 разделяют адресное пространство на четыре части (см. рис.1.3). Остается составить карту памяти исходя из логики работы входов разрешения, запрограммировать все области ПЗУ и проверить работу схемы на всех возможных режимах.

Проверку работоспособности запрограммированной схемы необходимо осуществлять при наличии на одном экране схемы ПЗУ и схемы дешифратора и управлении входами обоих схем одноименными источниками входных воздействий. Выходы обоих ИМС должны изменяться одинаково.

В отчете представить:

1. УГО дешифратора с обозначением используемых входов ПЗУ для моделирования его работы;
- 2 Схематическую карту программирования ПЗУ.

### Задание 1.3. Построить дешифратор для семисегментного индикатора

Семисегментный индикатор представляет собой полупроводниковый прибор, состоящий из отдельных сегментов и позволяющий с их помощью

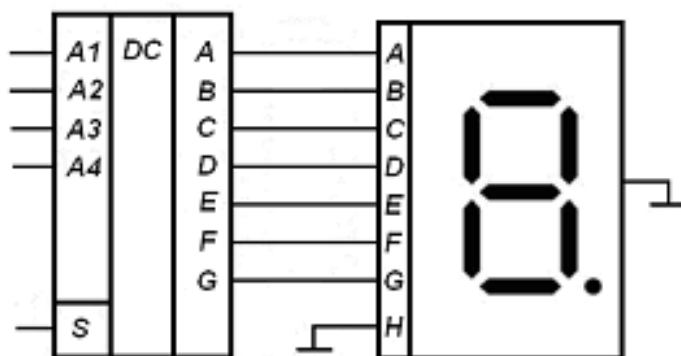


Рис.1.6. Подключение дешифратора к семисегментному индикатору

визуализировать символы заданного алфавита. Сегменты загораются при подаче высокого потенциала (единицы), управляются каждый автономно, но выдача сразу нескольких единиц образует комбинацию горящих сегментов, представляющую один символ алфавита. Подключение дешифратора к семисегментному индикатору показано на рис.1.6.

Построение дешифратора будет осуществляться на ИМС K155PE3.

Дешифратор с таким применением свое название носит условно. Он должен выдавать одновременно несколько выходных сигналов, что обычный дешифратор сделать не может. Это название закрепилось исторически. Фактически же в этом узле последовательно реализованы схемы дешифратора и шифратора.

Для выполнения задания необходимо реализовать следующий алгоритм:

1. Выбрать семисегментный индикатор из библиотеки или синтезировать его из рядом стоящих световых индикаторов при отсутствии такового;
2. Определить количество входов и выходов ПЗУ, необходимых для построения дешифратора. На индикаторе необходимо визуализировать арабские цифры от 0 до 9, задействовав одновременно несколько сегментов;
3. Определить количество слов, необходимых для создания дешифратора. Если индикатор предназначен для индикации арабских цифр, то потребуется 10 кодовых комбинаций, чтобы составить список слов для цифр от 0 до 9. Такой набор кодовых комбинаций часто называют знакогенератором;
4. Подключить выходы ПЗУ к сегментам индикатора. Для удобства программирования подключать выходы лучше последовательно по порядку нумерации сегментов. Такой вариант показан на рис.1.7;
5. Изменяя содержимое ПЗУ, создать знакогенератор арабских цифр. Для этого в каждой ячейке, выбираемой по соответствующему входному коду (адресу), необходимо запрограммировать единицы в тех битах, которые сопоставлены сегментам, образующим отображаемую цифру;
6. Подав активный сигнал на вход разрешения ПЗУ и задавая различные кодовые комбинации на адресные входы, убедиться в правильности функционирования дешифратора.

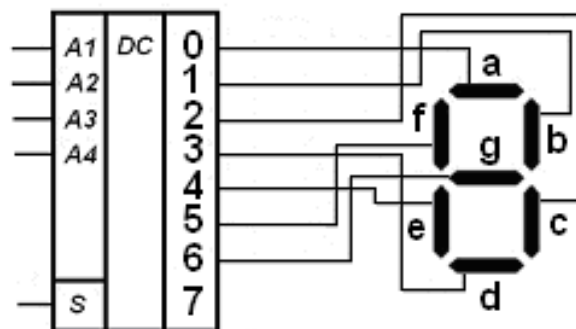


Рис.1.7. Схема подключения сегментов индикатора

В отчете представить:

1. УГО дешифратора с подключенным семисегментным индикатором;
2. Карту программирования ПЗУ.

#### **Задание 1.4. Построить полный двухразрядный сумматор на основе ПЗУ**

Полный сумматор характеризуется наличием входов слагаемых A2, A1 и B2, B1 и входом переноса (входом распространения переноса) CRP или CR

(от англ. CaRry – нести и Propagation – распространение), а также выходом суммы S2, S1 и выходом переноса (выходом генерации переноса) CRG или CG (от англ. CaRry Generation – выход генерации переноса). Его можно использовать в качестве элемента для конструирования многоразрядных сумматоров.

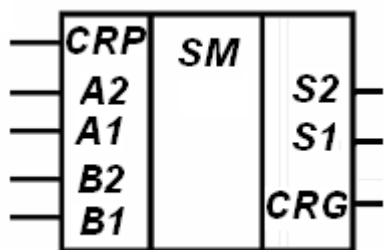


Рис.1.8. УГО 2-разрядного сумматора

УГО двухразрядного сумматора представлено на рис.1.8.

Эта задача решается в рамках методики синтеза комбинационных устройств на ПЗУ. Методика предполагает, что адресные входы ПЗУ будут играть роль входов синтезируемого устройства, а выбираемая при этом ячейка памяти должно содержать необходимые выходные сигналы для подаваемого набора входных переменных.

1. Предварительно следует определиться с количеством и назначением входов создаваемого сумматора.

2. Затем разработать таблицу истинности сумматора по формату:

CRP	A2	A1	B2	B1	CRG	S2	S1

Следует обратить внимание, что слагаемые имеют вес разряда и порядок разрядов важен. Поэтому для удобства составления таблицы переносы и разряды необходимо расположить так, как указано в таблице.

При полном переборе всех комбинаций входных сигналов, следуя логике арифметического сложения, необходимо рассчитать выходные значения функций суммы и переноса.

3. Рассматривая входные сигналы сумматора в качестве адресных входов ПЗУ, запрограммировать накопитель.

4. Проверить работоспособность сумматора на операциях

$$2+1=3;$$

$$3-2=1.$$

5. Организовать 6-разрядный сумматор, использовав несколько экземпляров запрограммированной схемы ПЗУ.

6. Выполнить вычисления по индивидуальному варианту в соответствии с номером по журналу учебной группы:

Номер по журналу учебной группы	Сложение	Вычитание
1, 15	15+7	27-13
2, 17	17+8	25-14
3, 16	18+9	22-15

Номер по журналу учебной группы	Сложение	Вычитание
6, 20	13+12	24-6
7, 19	12+14	23-12
8, 12	11+20	28-16

4, 13	19+10	20-11
5, 11	14+11	26-17

9, 14	10+15	29-21
10, 18	9+17	21-7

В отчете представить:

1. УГО 2-разрядного сумматора с обозначением входов ПЗУ, которые использованы при разработке устройства;
2. Таблицу истинности 2-разрядного сумматора (карту программирования ПЗУ);
3. Схему 6-разрядного сумматора;
4. Двоичные коды, использованные для выполнения всех вычислений.

### **Задание 1.5. Построить на ПЗУ неполный дешифратор с прямыми выходами с выборкой по заданным адресам**

Адреса для дешифрации выбираются по индивидуальным вариантам в соответствии с номером по журналу учебной группы:

Номер по журналу учебной группы	Выбираемые адреса
1, 14	15, 7, 27, 13, 17, 5
2, 16	17, 8, 25, 14, 9, 1
3, 15	18, 9, 22, 15, 6, 12
4, 12	19, 10, 20, 11, 4, 15
5, 20	14, 11, 26, 17, 3, 30

Номер по журналу учебной группы	Выбираемые адреса
6, 19	13, 12, 24, 6, 20, 2
7, 18	12, 14, 23, 12, 7, 31
8, 11	11, 20, 28, 16, 8, 17
9, 13	10, 15, 29, 21, 5, 16
10, 17	9, 17, 21, 7, 29, 14

Задача решается в соответствии общей методикой синтеза комбинационных схем. Однако необходимо учесть изменившееся состояние активных сигналов на выходе дешифратора, а также решить, что делать с входом разрешения ПЗУ. Кроме того, требуется принять решение по остальным комбинациям адресных входов при условии, что выборке подлежат только указанные в задании объекты.

Порядок решения следующий:

1. Перевести десятичные адреса в HEX-формат;
2. Определить количество необходимых адресных разрядов;
3. Изобразить УГО дешифратора с обозначением входов ПЗУ, которые будут их имитировать;
4. Разработать карту распределения памяти ПЗУ;
5. Запрограммировать схему ПЗУ;
6. Протестировать получившееся устройство на всех возможных наборах.

***В отчете представить:***

1. УГО разрабатываемого дешифратора с обозначением входов ПЗУ, которые использованы при разработке устройства и обозначением выходов в HEX-формате;
2. Карту распределения памяти ПЗУ.

***Вопросы для самопроверки***

1. Сформулируйте назначение входов и выходов микросхемы памяти 155РЕЗ.
2. Перечислите способы записи и стирания информации в ПЗУ.
3. Разработать индикатор телевизионных каналов на базе ПЗУ.
4. Разработать дешифратор на ПЗУ 155РЕЗ, тип выходов которого (прямые или инверсные) выбирается управляющим сигналом.
5. В чем состоят особенности микросхем динамической памяти?
6. Какие сигналы присутствуют на выходах микросхем 573РФ2 и 155РЕЗ при пассивных сигналах на входах выбора микросхемы?
7. Как обозначаются на корпусах различные типы микросхем?
8. Как, помимо программирования и управления старшим адресным разрядом можно изменить направление движения бегущей строки?
9. Сформулируйте на уровне структурной схемы предложения по созданию информационного табло по принципу бегущей строки, позволяющего отображать произвольную текстовую и числовую информацию.
10. Нарисуйте структурную схему ПЗУ. Каково назначение основных структурных элементов? Каково назначение входов и выходов ПЗУ?
11. Назовите основные параметры ПЗУ, охарактеризуйте их физический смысл.
12. Какие типы выходов применяются в микросхемах ПЗУ? Для чего их используют?
13. Какое состояние имеют элементы памяти ПЗУ до программирования? Как это определить?
14. Каково назначение входов и выходов ИМС 155РЕЗ? Каково ее условное графическое обозначение на схемах?
15. Каков порядок считывания данных из ИМС ПЗУ?
16. Для каких целей используют ПЗУ?

## 2. ИСПОЛЬЗОВАНИЕ ПЗУ В ДИНАМИЧЕСКИХ УСТРОЙСТВАХ

### Цель

1. Выработать практические умения синтеза динамических схем с использованием ПЗУ.
2. Совершенствовать практические умения и навыки работы с цифровой техникой, ЭВМ и системами моделирования.
3. Совершенствовать навыки анализа, обобщения и систематизации полученных результатов, навыки составления и оформления отчетных материалов, навыки точного и лаконичного представления докладов на вопросы технического характера.

### Учебные вопросы

- 2.1. Построить генератор сигнала бедствия, запускающийся по включению питания.
- 2.2. Создать бегущую строку из четырех цифровых индикаторов.
- 2.3. Создать бегущую строку с изменяющейся информацией.
- 2.4. Исследовать возможность увеличения длины бегущей строки до 8 индикаторов

### Литература для подготовки к занятию

1. Угрюмов Е.П. Цифровая Схемотехника: Учебное пособие для вузов. – СПб.: БХВ-Петербург, 2007. – 800с.
2. Бойко В.И. и др. Схемотехника электронных систем. Цифровые устройства.- СПб.: БХВ-Петербург, 2004.-512с.
3. Потемкин И.С. Функциональные узлы цифровой автоматики. - М.: Энергоатомиздат, 1988.-320с.
4. Схемотехника БИС постоянных запоминающих устройств. /О.А.Петросян, И.Я. Козырь и др.-М.: Радио и связь, 1987.-304с.
5. Полупроводниковые БИС запоминающих устройств: Справочник /Под ред. А.Ю. Гордонова и Ю.А. Дьякова. - М.: Радио и связь, 1987.-360с.
6. Лебедев О.Н. Микросхемы памяти и их применение.- М.: Радио и связь, 1990.-160с.
7. Шило В.Л. Популярные цифровые микросхемы. – М.: Радио и связь, 1987.-352с.

### Содержание отчета

1. Название работы.
2. Последовательно для каждого из учебных вопросов: название учебного вопроса и перечень тех материалов, которые указаны в конкретном задании;
3. Краткие ответы на те контрольные вопросы для самопроверки, которые ещё не нашли своего отражения в отчете.



### **Вопросы для самопроверки**

1. Нарисуйте структурную схему ПЗУ. Каково назначение основных структурных элементов? Каково назначение входов и выходов ПЗУ?
2. Назовите основные параметры ПЗУ, охарактеризуйте их физический смысл.
3. Какие типы выходов применяются в микросхемах ПЗУ? Для чего их используют?
4. Какое состояние имеют элементы памяти ПЗУ до программирования? Как это определить?
5. Каково назначение входов и выходов ИМС 155РЕЗ? Каково ее условное графическое обозначение на схемах?
6. Каков порядок считывания данных из ИМС ПЗУ?
7. Для каких целей используют ПЗУ?
8. Как в автоматическом режиме последовательно выбрать ячейки памяти?

### **Актуальность занятия**

В настоящее время широко используются информационные табло с оперативно меняющейся информацией и автономные несложные устройства оповещения и сигнализации. Многие из них используют механизм перебора адресов схем памяти с предварительно занесенной в нее информацией. Бегущие строки информационных табло используют, естественно, оперативную память, позволяющую менять отображаемые данные, но для ПЗУ механизм отображения совершенно идентичен. Поэтому для постоянных данных ограниченного размера сам механизм можно рассмотреть на основе схем К155РЕЗ.

### **Задание 2.1. Построить генератор сигнала бедствия, запускающийся по включению питания**

Генераторы с заданными заранее данными используются достаточно широко. Один из них это автономный световой или радиоэлектронный генератор сигнала бедствия. Запуск генератора должен осуществляться включением питания. Для моделирующих систем это требование легко обеспечить: фактически это соответствует включению режима моделирования.

Для данной задачи сигналы должны быть представлены кодовыми комбинациями, соответствующими, к примеру, символам S и O телеграфной азбуки. Если заполнить ячейки ПЗУ кодами этих символов в определенной последовательности и периодически передавать на выход ПЗУ с заданной скоростью содержимое слов, то можно имитировать подачу световых или радиосигналов бедствия "SOS" в виде телеграфных сообщений азбуки Морзе. При этом программировать необходимо длительность сигналов и длительность пауз в азбуке Морзе. Остается автоматизировать перебор адресов с заданной скоростью.

Применение счетчика в режиме счета как раз и позволяет осуществить автоматический перебор последовательно расположенных адресов ячеек памяти ПЗУ, если выходы счетчика использовать в качестве адресных сигналов ПЗУ. Включение в эту схему делителя частоты позволит задать необходимую скорость выдачи сигналов.

Функциональная схема такого автономного генератора представлена на рис.2.1.

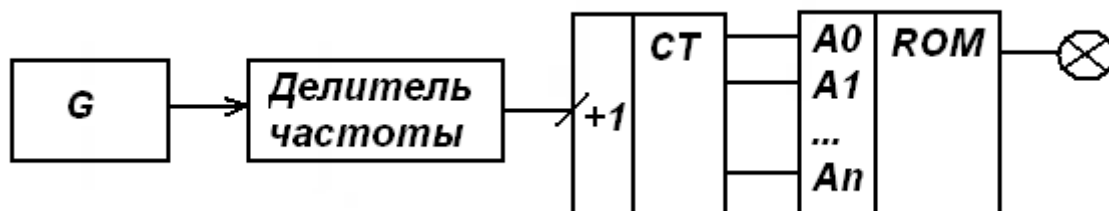


Рис.2.1. Функциональная схема автономного генератора

Алгоритм реализации задачи:

1. Получить данные о кодах символов сигнала бедствия в азбуке Морзе;
2. Рассчитать количество тактов генерации в одном периоде с учетом паузы между периодами. Определить требуемое число адресных входов ПЗУ;
3. Разработать функциональную схему с учетом конкретных рассчитанных данных. Обозначить сигналы на константных входах микросхем. Частоту генератора подобрать экспериментально, опираясь на возможности моделирующей программы, без использования делителя частоты;
4. Составить карту памяти ПЗУ;
5. Запрограммировать схему ПЗУ;
6. Проверить работоспособность схемы.

В отчете представить:

1. Функциональную схему устройства с конкретными рассчитанными параметрами;
2. Карту памяти ПЗУ для генерации сигнала бедствия.

## **Задание 2.2. Создать бегущую строку из четырех цифровых индикаторов**

Идея последовательного перебора адресов ПЗУ и, соответственно, последовательного отображения на выходе различных двоичных кодов является достаточно продуктивной. Если двоичные коды отображать соответствующими символами, а для вывода индикации применить несколько индикаторов, то запрограммировав и соответствующим образом включив ПЗУ, можно получить эффект бегущей строки.

Воспользуемся в качестве индикатора одной из схем, состоящей из дешифратора для 7-сегментного индикатора и собственно самого индикатора. Модель такой схемы есть в моделирующей программе и может быть вызвана на рабочий экран клавишей "I". Она имеет четыре входа и, соответственно, может отобразить все цифры шестнадцатеричной системы счисления.

Схему построить по индивидуальному варианту в соответствии с номером по журналу учебной группы:

Номер по журналу учебной группы	Направление движения	Комбинация символов
1, 14	Влево	A235
2, 16	Вправо	B367
3, 15	Влево	C478
4, 12	Вправо	D589
5, 19	Влево	E035

Номер по журналу учебной группы	Направление движения	Комбинация символов
6, 17	Влево	F279
7, 18	Вправо	A975
8, 11	Влево	D694
9, 13	Вправо	F952
10, 20	Влево	C753

Уточним теперь поставленную задачу. Требуется на основе ПЗУ построить устройство, реализующее эффект бегущей строки таким образом, чтобы на четырех последовательно расположенных 7-сегментных индикаторах осуществлялось движение четырех цифр 1234 (реально – по варианту). Направление движения – по варианту, сдвиг - автоматически, начало движения - подача питания. Исходное положение – строка пуста (нули, т.к. полное гашение индикатора моделью не предусмотрено).

Предлагается использовать две одинаково запрограммированных ИМС ПЗУ, к которым подключены четыре индикатора по четырем связям каждый. Получается, что четыре младших разряда каждой ячейки управляют одним индикатором, а четыре старших разряда той же ячейки – другим индикатором. При этом коды адресов обоих ПЗУ в каждый момент времени одинаковы и перебираются синхронно.

Каждый индикатор должен выводить цифры в следующей последовательности: 1, 2, 3, 4. Причем если на 1-м индикаторе высвечена цифра 1, то на 2-м индикаторе в это же время – цифра 2, на 3-м цифра 3 и т.д. Легко видеть, что каждый индикатор реализует свое собственное поведение, отображая одни и те же коды, но со сдвигом на один такт. Сдвиг на один такт - это размещение нужного кода в следующем последовательном адресе ПЗУ.

Реализация автономного поведения каждого индикатора (а их четыре) требует наличия самостоятельной области памяти (т.е. четырех областей). Для ПЗУ K155PE3 это возможно (см. рис.1.3). Однако при условии, что один байт ПЗУ управляет сразу двумя индикаторами, делить адресное пространство можно только на две половины. Делается это, как известно, старшим адресным входом. Микросхемы ПЗУ можно программировать одинаково, но

использовать разные области памяти для разных пар индикаторов, выделив их различными константами на старшем адресном входе. Тогда движение, например, кода четверки можно проиллюстрировать на ри.2.2.

ПЗУ1				ПЗУ2			
A0	0	0	0	0	0	0	0
A1	4	0	0	0	0	0	0
A2	0	4	0	0	0	0	0
A3	0	0	4	4	0	0	0
A4	0	0	0	0	4	4	4
0				1			

Рис.2.2. Движение кода "4" слева направо

На рис.2.2. каждая цифра соответствует коду из четырех разрядов байта. Соответственно левые две цифры кодируются одним байтом ПЗУ1, выбираемым по сигналам четырех адресных входов А0-А3 при константе 0 на входе А4, а левые две цифры – соответственно при том же адресном коде, но при константе 1 на входе А4 ПЗУ2. Всего на рисунке показано пять последовательно увеличивающихся адресных комбинаций. За это время код "4" переместился с левой границы в правую сторону до четвертого индикатора, отработав 5 тактов (устройство отобразило пять последовательных адресных комбинации). Всего адреса А0-А3 могут реализовать 16 тактов.

Последовательность выполнения задания предлагается следующая:

1. Представить символы бегущей строки в двоичном формате;
2. Разработать схему устройства, подключив к двум одинаковым ПЗУ индикаторы и обеспечив синхронный последовательный перебор адресов обоих накопителей (см. рис.2.1). Эта схема конкретизирует те полубайты, которые управляют каждым индикатором. Выбрать константы для А4 каждой микросхемы;
3. Составить карту памяти микросхемы ПЗУ. Особое внимание обратить на выполненное подключение каждого индикатора и требуемую для него динамику;
4. Запрограммировать ИМС ПЗУ;
5. Выполнить проверку на функционирование работы устройства.

В отчете представить:

1. Функциональную схему устройства с заданными константами;
2. Карту памяти ПЗУ для генерации эффекта бегущей строки.

### **Задание 2.3. Создать бегущую строку с изменяющейся информацией**

Изменение информации выбрать самостоятельно. В качестве рабочего варианта можно рассматривать обратную последовательность символов из задания 2.2. Смена информации должна происходить по командам пользователя.

Последовательность выполнения задания и состав отчета для этого задания аналогичны заданию 2.2. Если выполнить настоящее задание на двух одинаково запрограммированных ИМС ПЗУ не получится, то необходимо изучить возможность использования двух ИМС, запрограммированных по-разному.

### **Задание 2.4. Исследовать возможность увеличения длины бегущей строки до 8 индикаторов**

Решение этой задачи предполагает исходные данные, указанные в задании 2.2, одно направление движения, использование произвольного количества ИМС ПЗУ с двумя вариантами программирования.

Последовательность выполнения задания и состав отчета для этого задания аналогичны заданию 2.2.

#### **Задание для самопроверки**

1. Сформулируйте назначение входов и выходов микросхемы памяти 155РЕЗ.
2. Перечислите способы записи и стирания информации в ПЗУ.
3. Разработать индикатор телевизионных каналов на базе ПЗУ.
4. Разработать дешифратор на ПЗУ 155РЕЗ, тип выходов которого (прямые или инверсные) выбирается управляющим сигналом.
5. В чем состоят особенности микросхем динамической памяти?
6. Какие сигналы присутствуют на выходах микросхем 573РФ2 и 155РЕЗ при пассивных сигналах на входах выбора микросхемы?
7. Как обозначаются на корпусах различные типы микросхем?
8. Как, помимо программирования и управления старшим адресным разрядом можно изменить направление движения бегущей строки?
9. Сформулируйте предложения по созданию информационного табло по принципу бегущей строки, позволяющего отображать произвольную текстовую и числовую информацию.

### 3. РАЗРАБОТКА МОДУЛЯ ПАМЯТИ

#### Цель

1. Выработать практические навыки синтеза схем модулей памяти вычислительных систем по заданным параметрам.
2. Совершенствовать практические умения и навыки работы с цифровой техникой, ЭВМ и системами моделирования.
3. Совершенствовать навыки анализа, обобщения и систематизации полученных результатов, навыки составления и оформления отчетных материалов, навыки точного и лаконичного представления докладов на вопросы технического характера.

#### Учебные вопросы

- 3.1. Построение блока дешифрации адресов.
- 3.2. Построение схем дешифраторов адресов ОЗУ и ПЗУ
- 3.3. Разработка блока дешифрации адресов устройств ввода-вывода
- 3.4. Построение функциональной схемы модуля памяти.

#### Литература для подготовки к занятию

1. Чикалов А.Н. Микропроцессорные системы. Методические указания по выполнению курсового проекта. Ростов-на-Дону: Северо-Кавказский филиал МТУСИ, 2019.- 44с.
2. Микропроцессоры и микропроцессорные комплекты интегральных микросхем: Справочник: В 2 т. / Н.Н. Аверьянов и др.; Под ред. В.А. Шахнова. - М.: Радио и связь, 1988.
3. Микропроцессорный комплект К1810: Структура, программирование, применение: Справочная книга / Ю.М. Казаринов, В.Н. Номоконов, ГС. Подклетнов, Ф.В. Филиппов; Под ред. Ю.М. Казаринова. - М.; Высшая школа, 1990. - 269 с.
4. Лю Ю-Чжен, Гибсон Г. Микропроцессоры семейства 8086/8088. Архитектура, программирование и проектирование микропроцессорных систем. - М.: Радио и связь, 1987. - 512с.
5. Чикалов А.Н. Микропроцессоры. Периферийные устройства. Методические указания к лабораторным и практическим занятиям. Ростов-на-Дону: Северо-Кавказский филиал МТУСИ, 2019.- 37 с.
6. Жуковский А.Г., Манин А.А. Руководство по подготовке курсовых работ (проектов) и выпускных квалификационных работ. Ростов-на-Дону: СКФ МТУСИ, 2019. - 61с.

#### Содержание отчета

1. Название работы.
2. Последовательно для каждого из учебных вопросов: название учебного вопроса и перечень тех материалов, которые указаны в конкретном задании;

3. Краткие ответы на те контрольные вопросы для самопроверки, которые ещё не нашли своего отражения в отчете.

### **Вопросы для самопроверки**

1. Нарисуйте структурную схему ПЗУ. Каково назначение основных структурных элементов? Каково назначение входов и выходов ПЗУ?
2. Какие типы выходов применяются в микросхемах ПЗУ? Для чего их используют?
3. Каков порядок считывания данных из ИМС ПЗУ?
4. Для каких целей используют ПЗУ?
5. Основные выводы процессора, используемые для обращения к памяти и UVB.
6. В чем отличие совмещенного и раздельного адресного пространства?
7. В чем отличие схем статической и динамической памяти ИМС ОЗУ?

### **Актуальность занятия**

В современных вычислительных устройствах использование модулей памяти безальтернативно. Поэтому синтез схем, включающих конкретные объемы постоянной и оперативной памяти, является обязательным навыком проектировщиков.

### **Задание 3.1. Построение блока дешифрации адресов**

По исходным индивидуальным данным (по вариантам курсового проекта) провести расчет блоков дешифрации адресов постоянной и оперативной памяти для процессора семейства Intel x86.

Для разработки блока дешифрации адресов следует учесть структуру накопителей. В МПС необходимо обеспечить обмен как двухбайтными словами, так и отдельными байтами. Поэтому блок памяти следует организовать в виде двух 8-разрядных банков. Младший банк, хранящий данные D0-D7, должен иметь четные адреса обращения. Старший банк, хранящий данные D8-D15, должен иметь нечетные адреса обращения. Формировать тип передаваемых данных необходимо комбинациями сигналов адресного A0 и разрешения передачи старшего байта ВНЕ так, как указано в таблице 3.1. В соответствии с этой таблицей, 2-байтное слово должно располагаться начиная с четного адреса ( $A0=0$ ). В памяти в младшем адресе сохраняется младший байт 2-х байтного числа, а в старшем адресе - старший байт 2-х байтного числа. Адресом всего числа является адрес его самого младшего байта.

Таблица 3.1

Адресные комбинации для передачи байтов и слов  
по 16-разрядной шине данных МПС

ВНЕ	A0	Разрядность передаваемых данных
0	0	Все слово, оба байта по линиям D <sub>0</sub> ...D <sub>15</sub>
0	1	Старший байт по линиям D8...D15 (нечетный адрес)
1	0	Младший байт по линиям D0... D7 (четный адрес)
1	1	Нет передачи

Выравнивания многобайтных чисел по четным адресам не требуется. Если число начинается с четного адреса, то обмен осуществляется за одну операцию передачи по шине сразу 16 разрядов данных. Для многоразрядных чисел, начинающихся с нечетного адреса, обмен будет выполнен за два обращения к памяти, что снижает производительность процессора.

Сигналы A0 и ВНЕ должны подаваться на входы выбора соответственно младшего и старшего банка данных. Если потребуется страничная организация банков памяти, то удобно выбрать микросхемы памяти с двумя входами выбора для дополнительной селекции выбираемого банка данных. Такой вариант получится самым оптимальным. Структурная схема организации страницы памяти представлена на рис.3.1. Очевидно, что одна страница должна образовываться минимум двумя ИМС с байтной организацией.

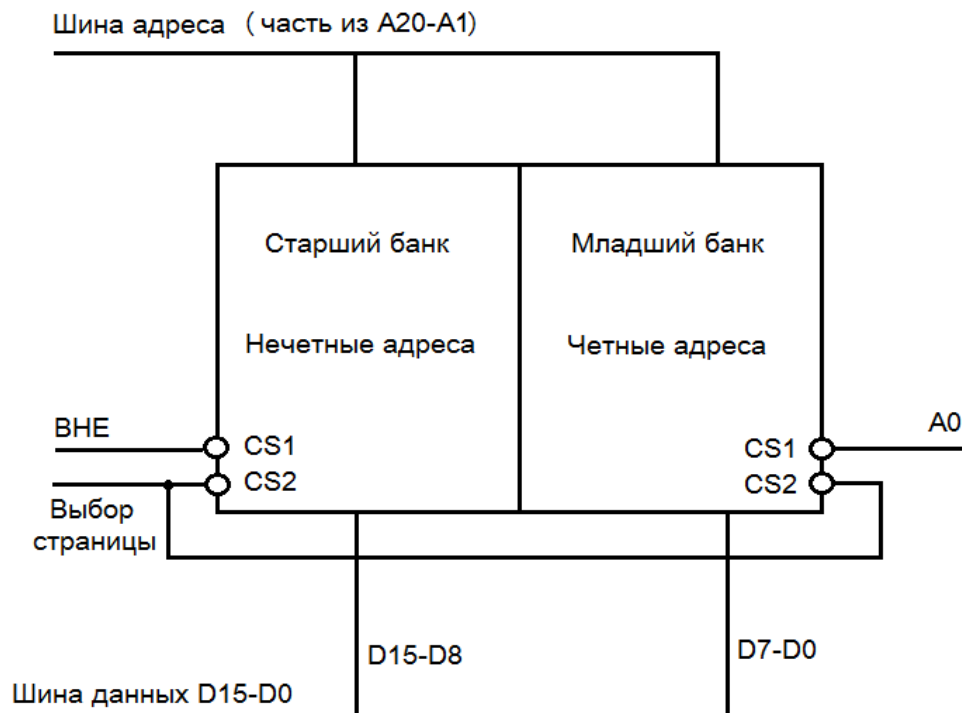


Рис.3.1. Структура страницы блока памяти

Блок дешифрации адресов проектируемой МПС состоит из трех частей: схемы формирования сигналов выбор кристалла (CS) для страниц (микро-



схем) ОЗУ блока памяти, схемы формирования сигналов CS для страниц (микросхем) ПЗУ блока памяти и схемы формирования сигналов CS для программируемых периферийных БИС блока внешнего интерфейса.

Исходными данными для разработки блока дешифрации адресов являются указанные в индивидуальном задании: начальный адрес ОЗУ, объем ОЗУ, тип микросхем ОЗУ, объем ПЗУ, тип микросхем ПЗУ, начальный адрес области портов ввода-вывода, а также разрядность системной шины данных, которая, в свою очередь, определяется заданным типом центрального процессора МПС. На начальном этапе разработки необходимо:

- 1) найти число БИС ОЗУ в блоке памяти;
- 2) найти число БИС ПЗУ в блоке памяти;
- 3) определить начальный адрес ПЗУ;
- 4) определить конечный адрес ОЗУ;
- 5) определить номера разрядов шины адреса для формирования общего сигнала выборки страниц ОЗУ ( $SELECT_{RAM}$ ) и общего сигнала выборки страниц ПЗУ ( $SELECT_{ROM}$ );
- 6) вычислить начальный адрес ввода-вывода для второй периферийной БИС (для первой он задан в задании).

Процесс разработки блока дешифрации адресов проиллюстрирован на следующем примере с использованием гипотетических типов ИМС.

Дано:

- 1) начальный адрес ОЗУ-38000H,
- 2) объем ОЗУ-32Kb,
- 3) БИС ОЗУ статического типа K537PYXX с организацией 8Kx1 (объемом 8Kбит),
- 4) объем ПЗУ-8Kb,
- 5) БИС ПЗУ типа K573PFX с организацией 2Kx8 (объемом 2Kb),
- 6) начальный адрес ввода-вывода для первой периферийной БИС таймера - 270H,
- 7) ЦП типа 8086 с 16-разрядной системной шиной данных.

Решение:

Условные графические обозначения ИМС, указанных в задании представлены на рис.3.2.

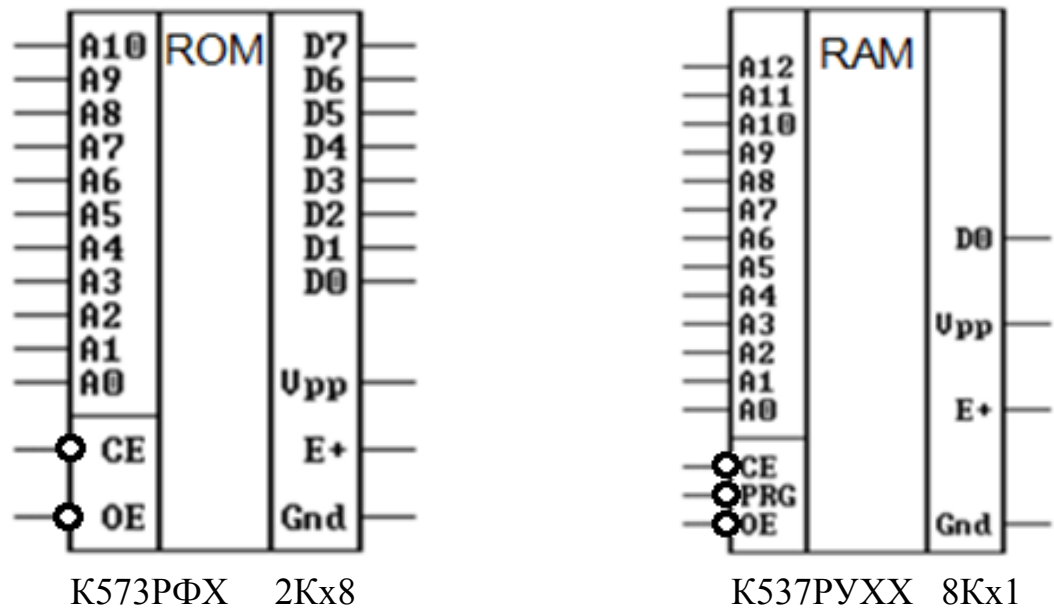


Рис.3.2. УГО ИМС памяти

У обоих ИМС имеется 2 входа разрешения с инверсным активным сигналом. Вход PRG у ИМС ПЗУ используется для программирования и в штатной работе процессора не применяется.

1). На основе исходных данных число банков в блоке памяти  $32\text{Kb}/8\text{Kb}=4$ , т.е. общий объем ОЗУ делится на объем одного банка. Объем банка определяется количеством адресуемых ячеек ИМС памяти. Если организация накопителя ОЗУ 8-разрядная, то число банков равно числу ИМС ОЗУ:  $N_{\text{BANK}}=N_{\text{RAM}}$ .

Однако если ИМС ОЗУ имеет другую организацию, то необходимо на основе имеющейся ИМС ОЗУ организовать банк с байтной структурой. Для рассматриваемого случая каждый банк необходимо создавать из восьми ИМС, соединив параллельно все ее входы. Такая схема представлена на рис.3.3. В этом случае выходы данных образуют единый байт с исходным количеством ячеек памяти. Поэтому в этом случае  $N_{\text{RAM}}=32$  при  $N_{\text{BANK}}=4$ .

С учетом принципов построения 16-разрядного блока памяти (см. рис.3.3) количество страниц равно двум.

2). Аналогично число БИС ПЗУ в блоке памяти  $N_{\text{ROM}} = 8\text{Kb}/2\text{Kb}=4$ . Для ПЗУ организация накопителя байтная, поэтому число блоков и число ИМС совпадают. Количество страниц для схем ПЗУ равно двум.

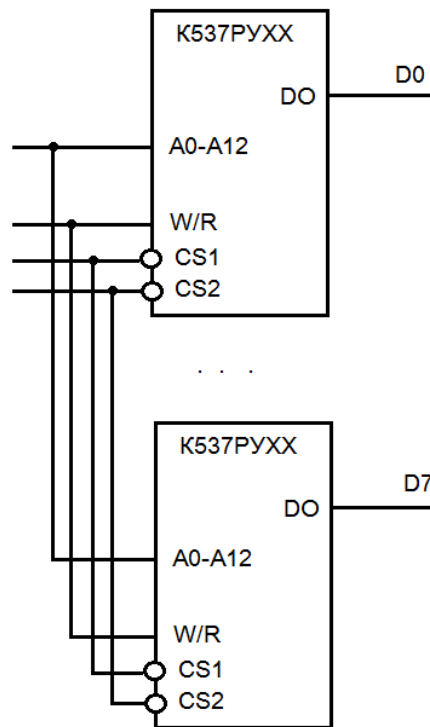


Рис.3.3. Организация байтной структуры банка памяти

3). Для определения начального адреса ПЗУ необходимо из всего объема адресного пространства процессора в 1Mb (100000H) вычесть заданный объем ПЗУ (число ячеек), т.е. 8Kb ( $8 * 2^{10} = 2000H$ ). Получается:

$$100000H - 2000H = FE000H.$$

Для вычисления шестнадцатеричного кода числа 8K достаточно понимать, что 1K образуется из 10 двоичных разрядов: 100 0000 0000. Чтобы убедиться в этом, достаточно проследить веса разрядов по мере смещения в сторону старших разрядов: последний ноль имеет вес 512. Следовательно, следующий разряд с единицей имеет вес 1024. Такое число и обозначается как 1K. Это число в шестнадцатеричной системе записывается как 400H. Однако дальнейшие преобразования выполнять в шестнадцатеричной системе довольно трудно, поэтому лучше продолжать работать в двоичной системе.

Для получения 8K необходимо 1K умножить на 8, что соответствует трем сдвигам влево. Поэтому 8K эквивалентно числу 10 0000 0000 0000. Переведя это число в шестнадцатеричную систему (каждую тетраду представив одной шестнадцатеричной цифрой), получим как раз 2000H.

Следует заметить, что умножение в шестнадцатеричной системе даст такой же результат. Если 400H умножить на 8, то должны получить впереди десятичное число 32. Но в шестнадцатеричной системе это число 20H. Поэтому итоговый результат получается такой же: 2000H.

ПЗУ всегда находится в самом конце первого мегабайта адресного пространства, т.к. после включения питания ЦП выбирает для исполнения первую команду по адресу FFFF0H. Это справедливо и для процессора 80286.

4). Для определения конечного адреса ОЗУ необходимо к заданному начальному адресу (38000H) прибавить число ячеек ОЗУ (объем ОЗУ для рассматриваемого примера - 32Kb или  $32 * 2^{10} = 8000H$ ) и вычесть единицу. Получаем:

$$38000H + 8000H - 1 = 3FFFFH.$$

5). Определить номера разрядов шины адреса для формирования общего сигнала выборки ОЗУ ( $SELECT_{RAM}$ ). Это те из 20-ти разрядов адреса, двоичные значения которых остаются постоянными для всех адресов из диапазона ОЗУ: 38000H ... 3FFFFH. Нумеруются разряды шины адреса соответственно A19 ... A0 слева направо.

$$\begin{aligned} 38000H &= \underline{0011} \underline{1000} \underline{0000} \underline{0000} \underline{0000}_2 \\ 3FFFFH &= \underline{0011} \underline{1111} \underline{1111} \underline{1111} \underline{1111}_2. \end{aligned}$$

Разряды, сохраняющие постоянное значение, подчеркнуты. Они соответствуют сигналам системной шины адреса A19 ... A15.

Таким образом, сигналы A19..A15 подаются на блок дешифрации адресов и формируют общий сигнал выборки ОЗУ:

$$SELECT_{RAM} = \bar{A}_{19} \& \bar{A}_{18} \& A_{17} \& A_{16} \& A_{15} \& (\overline{MRDC} \vee \overline{MWTC}).$$

В формировании сигнала  $SELECT_{RAM}$  участвуют также сигналы MRDC (сигнал чтения из памяти), MWTC (сигнал записи в память) системной шины управления, т.к. ОЗУ в блоке памяти должно быть выбрано (активно) только во время циклов чтения или записи в память. Во время циклов ввода-вывода ОЗУ не выбрано (пассивно; хранит ранее записанную информацию).

С помощью сигнала  $SELECT_{RAM}$  и нескольких старших разрядов из числа оставшихся - A14...A0 необходимо сформировать сигналы CS<sub>x</sub> для выборки отдельных страниц ОЗУ. Для заданного числа страниц (N<sub>STR</sub>) число дополнительных разрядов ША для дешифрации  $n = \log_2 N_{STR} = \log_2 2 = 1$ . Таким образом, для формирования отдельных сигналов CS1, CS2 должен использоваться разряд A14 системной шины адреса. Остальные 14 младших разрядов системной шины адреса (A13...A0) на блок дешифрации адресов не подаются, а подключаются непосредственно к адресным входам и входам CS микросхем ОЗУ в блоке памяти. Число сигналов CS в этом примере равно числу страниц ОЗУ, т.е. двум.

Перебирая все возможные комбинации для сигналов A14, запишем логические выражения для отдельных сигналов CS выборки страниц ОЗУ в рассматриваемом примере:

$$\begin{aligned} \overline{CS}_1 &= SELECT_{RAM} \& \bar{A}_{14} \\ \overline{CS}_2 &= SELECT_{RAM} \& A_{14}. \end{aligned}$$

В отчете представить:

1. Расчеты блоков дешифрации ПЗУ и ОЗУ по своему варианту;
2. Условные графические обозначения ИМС ПЗУ и ОЗУ по варианту.

### Задание 3.2. Построение схем дешифраторов адресов ОЗУ и ПЗУ

На основе имеющихся аналитических расчетов построить функциональные схемы блоков дешифрации адресов ОЗУ и ПЗУ.

Блок дешифрации адресов может строиться на базе отдельных логических элементов, микросхем дешифраторов, БИС ПЗУ или ПЛМ. Однако в блоке дешифрации адресов проектируемой МПС наиболее рационально формировать сигнал  $SELECT_{RAM}$  с помощью логических элементов и подавать его на вход выборки дешифратора, сигнал  $A_{14}$  - на адресные входы дешифратора, а сигналы  $CS_1, \dots, CS_2$  — снимать с выхода микросхемы дешифратора 2x4 с несколькими входами разрешения. Такая схема представлена на рис.3.4.

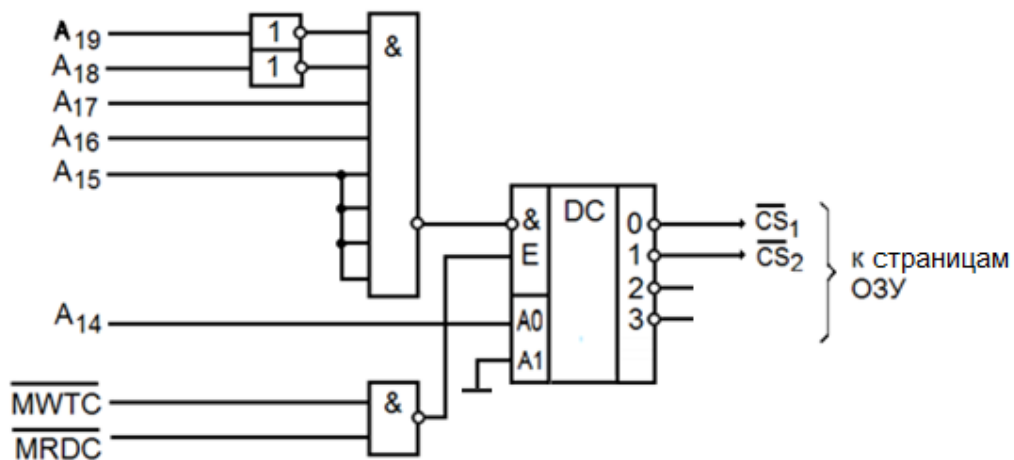


Рис.3.4. Схема дешифрации адреса ОЗУ

Аналогичные рассуждения позволяют получить логическое выражение для общего сигнала выборки ПЗУ:

$$FE000H = 1111\ 1110\ 0000\ 0000\ 0000$$

$$FFFFFH = \underline{1111\ 1111}\ 1111\ 1111\ 1111$$

$$SELECT_{ROM} = A_{19} \& A_{18} \& A_{17} \& A_{16} \& A_{15} \& A_{14} \& A_{13} \& \overline{MRDC}$$

Здесь используется только управляющий сигнал чтения из памяти  $MRDC$ , т.к. в процессе функционирования МПС из ПЗУ только считываются

коды машинных команд управляющей программы. При этом для формирования отдельных сигналов выборки страниц ПЗУ используется разряд A12 системной шины адреса. Требуется только 1 разряд, т.к. количество страниц ПЗУ как и ОЗУ равно двум. Остальные 12 младших разрядов системной шины адреса (A11...A0) на блок дешифрации адресов не подаются, а подключаются непосредственно к адресным входам микросхем ПЗУ в блоке памяти. Замечание относительно разрядности системной шины данных и организации памяти в равной мере справедливо и для ПЗУ.

На рис.3.5 для рассматриваемого примера показан вариант схемы формирования сигналов CS выборки страниц ПЗУ блока памяти.

6). Порты ввода-вывода адресуются 16-ю младшими разрядами системной шины адреса. В проектируемой МПС достаточно использовать лишь 10 младших разрядов (A9...A0), т.к. адреса схем начинаются только с адреса 270H. При этом два младших разряда подключается непосредственно к адресным входам периферийных БИС в блоке внешнего интерфейса. Таким образом, в блоке дешифрации адресов будут использоваться только 8 разрядов ША (A9...A2).

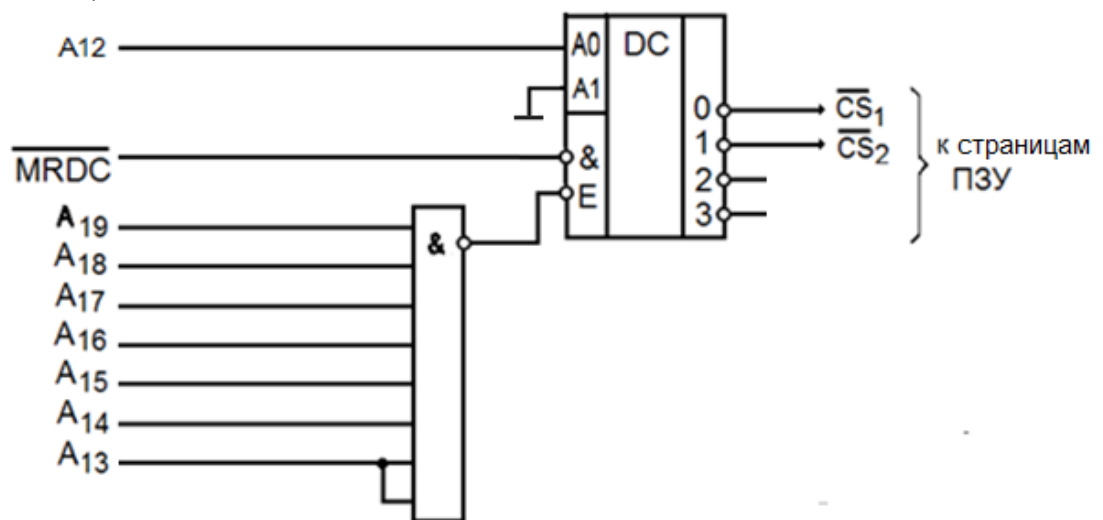


Рис.3.5. Схема дешифрации страниц блока ПЗУ

В отчете представить:

1. Функциональную схему блоков дешифрации адресов ОЗУ и ПЗУ.

### Задание 3.3. Разработка блока дешифрации адресов устройств ввода-вывода

Выполнить расчет блока адресов внешних устройств по индивидуальному заданию (по варианту задания курсового проекта) и построить функциональную схему дешифратора.

Для вариантов с 16-разрядной шиной данных обращение к внутренним регистрам периферийной БИС в управляющей программе выполняется только командами передачи байта по четному адресу из области ввода-вывода.

В этом случае сигнал  $A_0$  системной шины адреса для выбора портов не используется (и всегда равен нулю в четном значении адреса), а адресные входы периферийной БИС подключаются к системной шине адреса со сдвигом на один разряд, т.е. вход  $A_0$  к линии  $A_1$  шины, вход  $A_1$  к линии  $A_2$  шины и т.д.

Отсюда следует, что начальный адрес ввода-вывода для последующей периферийной БИС (если такой требуется) на 8 больше заданного начального адреса ввода-вывода для предыдущей БИС. То есть для рассматриваемого примера

$$\text{Addr.2} = \text{Addr.1} + 8H = 270H + 8H = 278H.$$

Представив начальные адреса в виде двоичных комбинаций адресных сигналов, получаем логические выражения для сигналов выборки двух периферийных БИС:

$$\begin{aligned}\overline{CS_1} &= A_9 \& \overline{A_8} \& \overline{A_7} \& A_6 \& A_5 \& A_4 \& \overline{A_3} \& (\overline{IORC} \vee \overline{IOWC}) \\ \overline{CS_2} &= A_9 \& \overline{A_8} \& \overline{A_7} \& A_6 \& A_5 \& A_4 \& A_3 \& (\overline{IORC} \vee \overline{IOWC})\end{aligned}$$

Адресная линия  $A_3$  позволяет различить две ИМС внешнего устройства.

Следует учесть, что адресные линии  $A_{15}$ - $A_{10}$  не учтены в выражениях для выбора внешних устройств (эти разряды в выражениях несущественны). Поэтому случайно использованные адреса в этом диапазоне разрядов будут активизировать внешние схемы. Обращаться по ним недопустимо.

В формировании сигналов выборки участвуют также сигналы  $IORC$ ,  $IOWC$  системной шины управления, т.к. каждая периферийная БИС в блоке внешнего интерфейса должна быть выбрана (для обмена данными между ЦП и ее внутренними регистрами) только во время циклов ввода-вывода. Во время циклов обращения к памяти программируемые периферийные БИС отключены от системной шины данных, т.е. соответствующие их выходы находятся в третьем состоянии (Z-состоянии).

На рис.3.6 для рассматриваемого примера показана схема формирования сигналов  $CS$  выборки двух периферийных БИС блока внешнего интерфейса. Схемы рис.3.4, 3.5 и 3.6. совместно образуют функциональную схему разработанного блока дешифрации адресов для приведенного примера. После выбора конкретных типов микросхем логики и дешифраторов на базе функциональной схемы строится принципиальная электрическая схема блока.

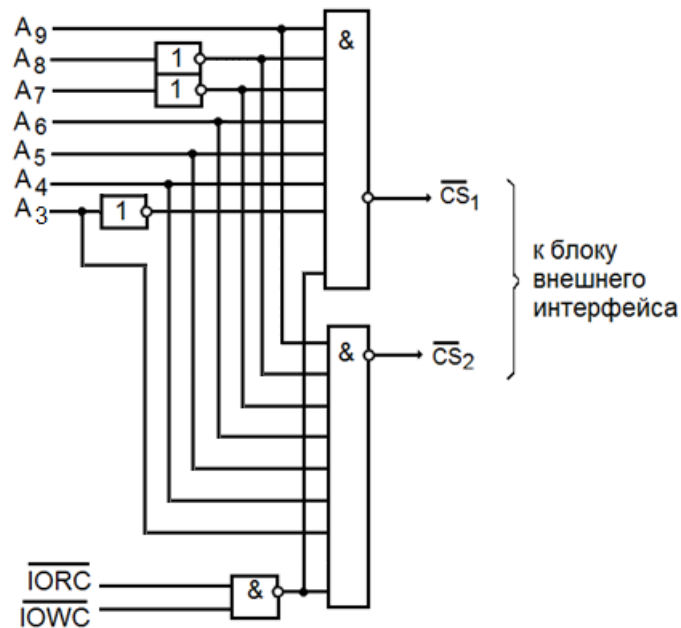


Рис.3.6.Схема дешифрации адреса периферийных БИС

В отчете представить:

1. Расчеты схемы дешифратора для внешних устройств;
2. Функциональную схему блоков дешифрации адресов внешних устройств.

### Задание 3.4. Построение функциональной схемы модуля памяти

На основании расчетов блоков дешифрации адресов ОЗУ и ПЗУ, характеристик ИМС ПЗУ и ОЗУ построить функциональную схему модуля памяти по индивидуальному заданию.

Наиболее просто блок памяти реализуется для вариантов с 8-разрядной системной шиной данных, в которых заданы БИС ОЗУ статического типа. В этом случае все одноименные входы-выходы D0...D7 всех БИС блока соединяются параллельно и подключаются к соответствующим линиям данных системной шины. Аналогично все одноименные входы адреса всех БИС блока соединяются параллельно и подключаются к соответствующим линиям адреса системной шины, не использованным в блоке дешифрации адресов. Отдельные сигналы CS, сформированные в блоке дешифрации адресов, подаются на входы CS соответствующих БИС памяти. Если микросхема памяти имеет несколько таких входов, все они соединяются параллельно, либо на дополнительные входы CS постоянно подается активный логический уровень. Все входы W/R БИС ОЗУ соединяются параллельно и подключаются к выходу MWTC системного контроллера, т.е. к линии сигнала "запись в память" системной шины управления.



Для вариантов с 16-разрядной системной шиной данных, в которых заданы БИС ОЗУ статического типа, блок памяти немного сложнее. Все БИС ОЗУ, как и все БИС ПЗУ, необходимо разбить попарно. Каждая пара, образуемая младшим и старшим банком данных, представляет собой страницу памяти. По одному из входов CS микросхемы в паре объединяются и на них подается соответствующий сигнал CS<sub>х</sub> выбора страницы из блока дешифрации адресов. Микросхемы в паре активизируются одновременно, только если в цикле чтения или записи, начиная с четного адреса, происходит передача слова, т.е. если в цикле используются все 16 линий данных системной шины. В других случаях выбирается только одна микросхема из пары. Логика выбора для различных циклов памяти при наличии 16-разрядной системной шины данных была отражена в табл.3.1.

Одна БИС из пары хранит байты по четным адресам и ее входы-выходы данных должны быть подключены к одноименным линиям D0 ... D7 шины данных. Другая БИС из пары хранит байты по нечетным адресам и ее входы-выходы данных должны быть подключены к линиям D8...D15 шины данных.

Если БИС ПЗУ и БИС ОЗУ статического типа имеют минимум два входа выборки CS, то вторые входы выборки кристалла необходимо использовать для индивидуальной активизации каждой БИС ОЗУ и ПЗУ. На второй инверсный вход CS 1-й микросхемы из пары (D0-D7, четные адреса) подается младший адресный сигнал системной шины - A0. А на второй инверсный вход CS 2-й микросхемы из пары (D8-D15, нечетные адреса) подается управляющий сигнал ВНЕ ("выборка старшего байта") от центрального процессора МПС. Все одноименные входы адреса всех БИС блока соединяются параллельно и подключаются к линиям адреса системной шины, не использованным в блоке дешифрации адресов, начиная с адресной линии системной шины A1. Другими словами, адресный вход A0 каждой микросхемы памяти подключается к линии A1 системной шины адреса, адресный вход A1 микросхемы памяти подключается к линии A2 системной шины и т.д.

Если ИМС памяти имеет один вход CS, то на нем по схеме логического умножения активных сигналов должны быть объединены сигнал CS блока дешифрации и сигнал A0 (для младшего банка четные адреса) и ВНЕ (для старшего банка нечетные адреса) соответственно.

Описанная схема включения (рис.3.7) обеспечивает необходимый порядок выборки микросхем ОЗУ и ПЗУ в блоке памяти со словной организа-

цией.

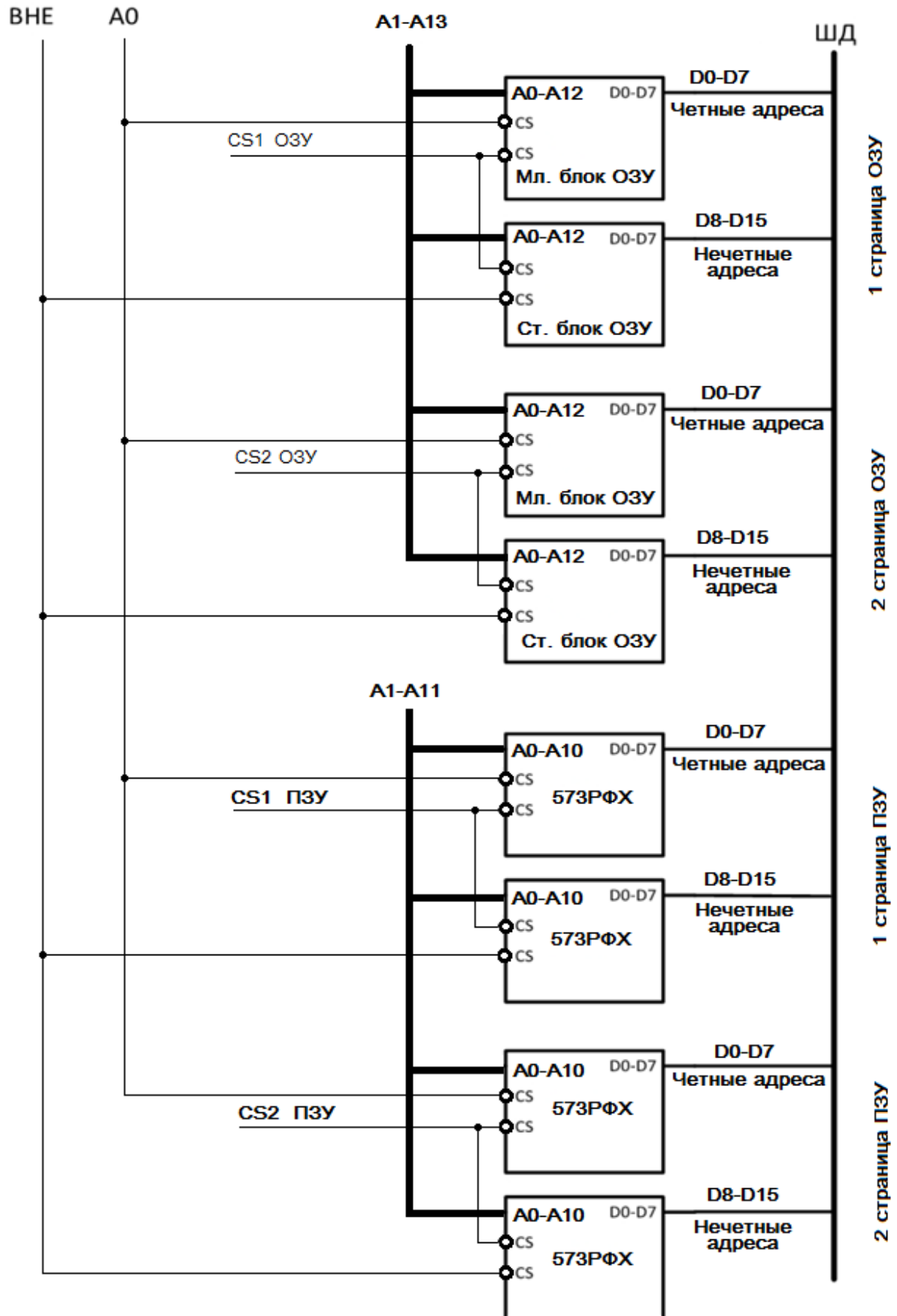


Рис.3.7. Вариант построения блока памяти ОЗУ и ПЗУ

Для вариантов с БИС ОЗУ динамического типа (заданный объем оперативной памяти - 64Kbit и более) при расчете числа микросхем необходимо пользоваться формулой:

$$N_{\text{RAM}} = (M_{\text{ОБЩ}}/M_1) \cdot 8,$$

где  $M_{\text{ОБЩ}}$  - заданный объем ОЗУ в килобайтах,

$M_1$  - число ячеек в одной БИС заданного типа в килобитах. БИС ОЗУ динамического типа имеют битовую организацию [5].

Для управления мультиплексированием адресных сигналов, формирования сигналов RAS (выбор строки) и CAS (выбор столбца), а также для организации режима регенерации микросхем ОЗУ динамического типа в блоке памяти целесообразно использовать специализированную БИС - контроллер динамического ОЗУ, например, отечественную микросхему K1810BT03 [2, раздел 5.5]. Типовые структурные схемы блока динамического ОЗУ с использованием БИС K1810BT03 можно найти в литературе [2, рис. 5,45 на с.190, рис. 5.47 на с.191].

#### Задание для самопроверки

1. Каково назначение основных входов и выходов в микросхемах памяти?
2. Каким образом увеличить разрядность шины данных в микросхемах памяти? Нарисуйте схему такого наращивания.
3. Каким образом увеличить число ячеек блока памяти? Нарисуйте схему такого наращивания.
4. Сформулируйте этапы синтеза дешифратора блока памяти.
5. Синтезируйте дешифратор с заданной областью адресов блока памяти.
6. Каким образом можно увеличить число выходов дешифратора адресов памяти?
7. Каким образом дешифратор подключается к сигнальным выводам микропроцессора?
8. Каков коэффициент расширения выводов шины адреса микропроцессора?
9. Каким образом увеличивают коэффициент расширения выводов адреса микропроцессора?
10. Как реализуются совмещенные и отдельные адресные пространства МП? Какие достоинства и недостатки имеют оба подхода?