

Вопросы по спецкурсу
«Микропрограммирование»
осень 2010

1. Понятие вычислительной системы (ВС). Задачи, решаемые ВС, классификация ВС по типам решаемых задач. Общая модель ВС, состав и назначение отдельных частей.
2. Производительность ВС, способы оценки производительности. Организации, производящие оценку производительности ВС. Связь производительности и стоимости ВС.
3. Требования к ВС по масштабируемости и совместимости ПО. Открытые системы.
4. Надежность и отказоустойчивость ВС. Дерево логических возможностей работы ВС, оценка надежности ВС.
5. Архитектура ВС, уровни архитектуры ВС.
6. Комбинационные логические устройства: мультиплексор/демультиплексор, шифратор/дешифратор.
7. Комбинационные логические устройства: цифровой компаратор, сумматоры.
8. Последовательностные логические устройства: триггеры, регистры.
9. Последовательностные логические устройства: счетчики, конечные автоматы.
10. Модель четырехразрядного АЛУ, режимы работы.
11. Типы управляющих (операционных) автоматов. Принцип микропрограммного управления. Микрокоманда, способы построения полей микрокоманд.
12. Память, способы классификации. СОЗУ. Виды и принцип работы ассоциативной памяти. Кэш-память процессоров.
13. Память, способы классификации. Виды и принцип работы ОЗУ.
14. Модели логической памяти процессоров. Способы преобразования логического адреса в линейный.
15. Модели виртуальной памяти процессоров. Способы преобразования линейного адреса в физический. Пример в случае архитектуры IA-32.
16. Принстонская и гарвардская архитектуры процессоров. Модификации гарвардской архитектуры. Гибридные архитектуры. Архитектуры системы команд процессоров. Классификация Флинна ВС. Примеры.
17. Модель нейрона, основные характеристики нейрона. Нейросеть. Применение нейросетей, персептрон. Нейропроцессоры.

Тематика задач

1. Написание программы на Ассемблере, решающей несложную вычислительную задачу.
2. Построение сокращенной нормальной формы булевой функции, заданной таблицей истинности и графическое изображение этой функции в виде цифровой схемы.
3. Построение конечного автомата, заданного в описательной форме или графом переходов.
4. Написание микропрограммы для заданного алгоритма.

Задачи по курсу «Микропрограммирование»

1. Написать программу на языке ассемблера определения количества четных чисел в последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.
2. Написать программу на языке ассемблера определения номера минимального числа в последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.
3. Написать программу на языке ассемблера определения разности сумм четных и нечетных элементов последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.
4. Написать программу на языке ассемблера определения максимального числа в последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.
5. Написать программу на языке ассемблера определения количества нечетных чисел в последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.
6. Написать программу на языке ассемблера определения количества чисел, меньших C_1 , в последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.
7. Написать программу на языке ассемблера определения максимального числа в последовательности слов C_1, C_2, \dots, C_N , расположенных последовательно в памяти. Логический адрес первого элемента последовательности и количество элементов N определяются константами. Результат вывести на экран.

1. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2ИЛИ-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	0	0100	1	1000	0	1100	1
0001	1	0101	0	1001	1	1101	1
0010	1	0110	1	1010	0	1110	1
0011	0	0111	1	1011	0	1111	1

2. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2И-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	0	0100	1	1000	0	1100	1
0001	1	0101	0	1001	1	1101	0
0010	1	0110	0	1010	0	1110	1
0011	0	0111	1	1011	1	1111	1

3. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2ИЛИ-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	1	0100	0	1000	1	1100	1

0001	1	0101	0	1001	1	1101	0
0010	1	0110	1	1010	0	1110	1
0011	0	0111	1	1011	1	1111	1

4. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2И-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	0	0100	1	1000	0	1100	0
0001	1	0101	0	1001	1	1101	1
0010	1	0110	0	1010	0	1110	1
0011	1	0111	1	1011	1	1111	0

5. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2ИЛИ-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	0	0100	1	1000	0	1100	1
0001	0	0101	0	1001	0	1101	1
0010	1	0110	0	1010	1	1110	1
0011	0	0111	0	1011	0	1111	1

6. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2И-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	1	0100	1	1000	1	1100	0
0001	1	0101	0	1001	1	1101	0
0010	1	0110	1	1010	0	1110	0
0011	0	0111	1	1011	1	1111	1

7. Построить сокращенную нормальную форму булевой функции $F(x_0, x_1, x_2, x_3)$, заданной таблицей истинности, в базисе 2ИЛИ-НЕ и нарисовать соответствующую ей комбинационную схему.

$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F	$x_3x_2x_1x_0$	F
0000	1	0100	1	1000	0	1100	0
0001	1	0101	0	1001	1	1101	0
0010	0	0110	0	1010	1	1110	1
0011	0	0111	1	1011	0	1111	1

1. Допустим, что одновременно в кабине лифта, вмещающей шесть человек, может подниматься не более трех человек. Если число пассажиров превышает три, то должен выдаваться сигнал на блокировку работы лифта (сигнал перегрузки). Входным сигналом является двоичный код, единица в нулевом разряде которого обозначает увеличение числа пассажиров на единицу, что фиксируется соответствующим датчиком, а единица в первом разряде – уменьшение пассажиров на единицу. Необходимо построить граф конечного автомата, описывающего работу контролирующего устройства лифта, выбрать количество и тип триггеров, запоминающих состояния автомата, и составить таблицу истинности, описывающую значения

управляющих сигналов входами триггеров в зависимости от текущего состояния автомата и значения входного сигнала, а также сигнал перегрузки.

2. Необходимо спроектировать конечный автомат с двумя режимами работы, управляемый входным сигналом M . При $M=0$ автомат работает как двоичный счетчик с модулем счета 8, при $M = 1$ как счетчик в коде Грея.

Десятичная цифра	Код Грея	Десятичная цифра	Код Грея
0	000	4	110
1	001	5	111
2	011	6	101
3	010	7	100

Постройте граф переходов конечного автомата, выберите количество и тип триггеров, запоминающих текущее состояние автомата, составьте таблицу истинности, описывающую значения управляющих сигналов триггеров в зависимости от текущего состояния автомата и сигнала M .

3. Необходимо спроектировать конечный автомат с двумя режимами работы, управляемый входным сигналом M и реализующий реверсивный счетчик Джонсона. При $M = 0$ автомат считает в «прямом» порядке (к номеру текущего состоянию при переходе прибавляется 1 по модулю 8), при $M = 1$ автомат считает в «обратном» порядке (от номера текущего состояния при переходе отнимается 1 по модулю 8). Соответствие номера состояния и выхода счетчика приведено в таблице.

Номер состояния	Код Джонсона	Номер состояния	Код Джонсона
0	0000	4	1111
1	0001	5	1110
2	0011	6	1100
3	0111	7	1000

Постройте граф переходов конечного автомата, выберите количество и тип триггеров, запоминающих текущее состояние автомата, составьте таблицу истинности, описывающую значения управляющих сигналов триггеров в зависимости от текущего состояния автомата и сигнала M , а также значения выходных сигналов.

4. Задан граф переходов конечного автомата, управляемого одним входным сигналом M и имеющего один выходной сигнал Z .

Выберите количество и тип триггеров, запоминающих текущее состояние автомата, составьте таблицу истинности, описывающую значения управляющих сигналов триггеров в зависимости от текущего состояния автомата и сигнала M , опишите булевы функции для каждого управляющего сигнала и выходного сигнала Z .

5. Задан граф переходов конечного автомата, управляемого одним входным сигналом M и имеющего один выходной сигнал Z .

Выберите количество и тип триггеров, запоминающих текущее состояние автомата, составьте таблицу истинности, описывающую значения управляющих сигналов триггеров в зависимости от текущего состояния автомата и сигнала M , опишите булевы функции для каждого управляющего сигнала и выходного сигнала Z .

1. Используя смешанный способ кодирования микроопераций и естественную адресацию для заданного в виде блок-схемы алгоритма составьте микропрограмму приведенного алгоритма.