



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**  
**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**  
**(ДГТУ)**

**Методические указания для выполнения контрольной работы**  
**по дисциплине**  
**«Компьютерные методы проектирования»**  
**для обучающихся по направлению подготовки**  
**15.04.04 Автоматизация технологических процессов и производств**  
**программа магистратуры**  
**«Автоматизация процессов и производств нефтегазового комплекса»**

Изучение курса «Компьютерные методы проектирования» включает:

- а) работу над учебными пособиями;
- б) выполнение контрольных работ;
- в) посещение лекций и консультации по отдельным разделам курса.

По основным вопросам и наиболее сложным темам курса, которые вызывают затруднения при самостоятельном изучении, студентам читаются лекции.

После изучения очередной темы курса студент должен уметь ответить на вопросы для самопроверки.

Контрольная работа включает в одно задание. Вариант задания определяется по порядковому номеру студента в журнале посещений/успеваемости.

После выполнения контрольной работы студент допускается к сдаче экзамена.

## Задание

1. Ознакомиться с логической схемой построения одноразрядного двоичного полусумматора и сумматора. Промоделировать схемы с использованием любого доступного пакета, например MC9 DEMO (или аналогичных) и составить таблицы истинности каждой схемы.

2. Разработать схему много разрядного сумматора последовательного переноса требуемой разрядности, согласно варианту (Таблица 1).

Таблица 1

	Разрядность устройства									
Вариант	1	2	3	4	5	6	7	8	9	10
Сумматор с последовательным	6	5	4	3	6	5	4	3	4	6
Сумматор с параллельным переносом	3	4	5	5	4	3	4	4	5	3
АЛУ	4	3	3	4	3	4	4	5	3	3
	Разрядность устройства									
Вариант	11	12	13	14	15	16	17	18	19	20
Сумматор с последовательным	3	6	5	3	2	2	4	4	3	5
Сумматор с параллельным переносом	4	4	4	2	4	2	2	2	3	3
АЛУ	5	4	2	3	4	3	3	4	3	5

3. Промоделировать работу сумматора, подав на его входы сигналы соответствующие двум слагаемым. Зарисовать временные диаграммы. Определить максимально возможное время суммирования, если задержка каждого простейшего логического элемента, составляющего сумматор, равна 10 нс. Подобрать такие сигналы на входах сумматора чтобы время суммирования было максимальным.



Заменить в схеме полусумматоры их логическими схемами и получить таблицу истинности аналогично, как полусумматора с использованием пакета Micro-Cap 9 DEMO. Логические выражения выходов сумматора будут иметь вид:

$$S = A \oplus B \oplus CI$$

$$CO = A \& B + (A \oplus B) \& CI = (1) \\ = Y + X \& CI$$

К п.2. Схема 3-х разрядного сумматора последовательного переноса приведена на рис.3. На сумматор подаются двоичные разряды слагаемых A0, A1, A2 и B0, B1, B2 (A0 и B0 – младшие разряды), а также бит переноса в младший разряд C0. Сумма получается на выходах S0, S1, S2, S3. Заменить в схеме сумматоры их логическими схемами, в отчете представить временную диаграмму, полученную при сложении двух чисел заданной по варианту разрядности.

Особенностью данного типа сумматора является большое время суммирования чисел, зависящее от разрядности сумматора. В наихудшем случае, если принять задержку любого простейшего логического элемента за единицу, время суммирования 3-разрядного сумматора будет  $K=7$ . Максимально возможное время суммирования будет равно:

$$T_{\text{сум}} = K * 10\text{нс}$$

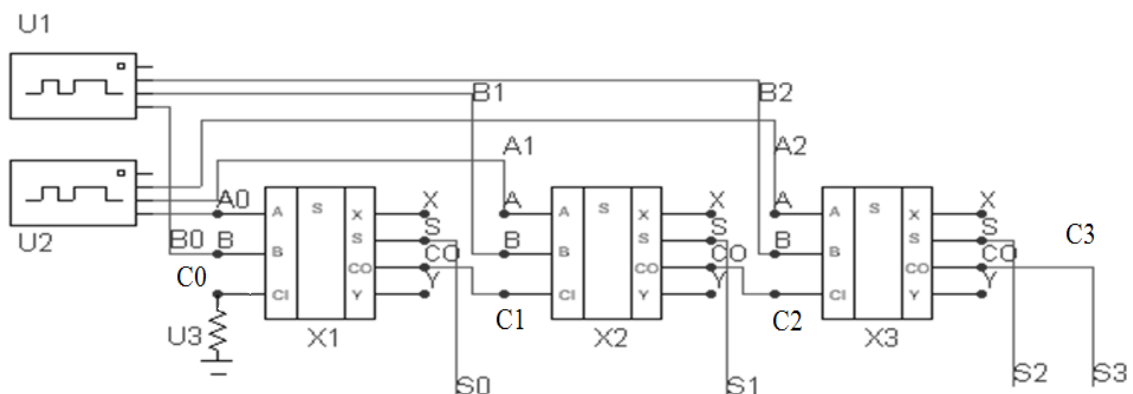


Рис.2. Схема сумматора последовательного переноса

При построении временной диаграммы с помощью Stimulus Generator требуемой разрядности выдать на сумматор слагаемые, в момент времени

отличный от начального. После этого фиксировать изменение сигналов суммы на всех N+1 выходах.

Примечание: Программирование работы Stimulus Generator заключается в задании двух основных параметров:

- Format задается в виде строки единиц их число равно числу выходов генератора (например: 1111 для генератора с четырьмя выходами);
- Command задается в виде последовательности двоичных кодов, которые будут выдаваться на выходы генератора в назначенные моменты времени (например: 0ns 0000 500ns 0110 ).

Для вывода временной диаграммы нужно воспользоваться командой Analysis-Probe Transient analysis, и после появления двух окон, указывая в одном окне на схеме точки, в которых необходимо проанализировать сигнал, во втором окне будут появляться соответствующие графики изменения логического сигнала (временная диаграмма).

Для определения максимально возможного времени суммирования, необходимо выделить все логические элементы схемы, выполнить двойной клик мышью на одном из них, задать в полях IPNL время задержки 10 нс (10ns). В отчете на временной диаграмме необходимо обозначить максимальное время суммирования и время формирования переноса.

К п.3. Перед построением сумматора с параллельным переносом, выведем формулы для сигналов переноса C0, C1, C2, C3:

$$C0 = C0$$

$$C1 = A0 \& B0 + (A0 \oplus B0) \& C0 = Y0 + X0 \& C0$$

$$C2 = A1 \& B1 + (A1 \oplus B1) \& C1 = Y1 + X1 \& C1 = Y1 + X1 \& Y0 + X1 \& X0 \& C0 \quad (2)$$

$$C3 = Y2 + X2 \& C2 = Y2 + X2 \& Y1 + X2 \& X1 \& Y0 + X2 \& X1 \& X0 \& C0$$

В полученных выражениях все переменные X0,...,X2 и Y0,...,Y2 могут быть вычислены за время равное задержки срабатывания одного логического элемента (10нс). В свою очередь, все логические слагаемые в выражениях вычисляются еще за один такой же интервал (10нс). И операция логического суммирования (ИЛИ), также осуществляется за 1 интервал. Таким образом, если сигналы переноса вычислять отдельно с помощью специальной

логической схемы в соответствии с (2) и заводить на сумматоры, то суммирование при любой разрядности будет выполняться за время равное 4 интервалам. Именно на этом принципе и построен сумматор с параллельным переносом.  $t_{\text{сум}} = 4\Delta t$

Схема пятиразрядного сумматора с параллельным переносом приведена на рис.3. Сигналы переноса C1, C2, C3, C4, C5 формируются с помощью схем выполненных на простейших логических элементах.

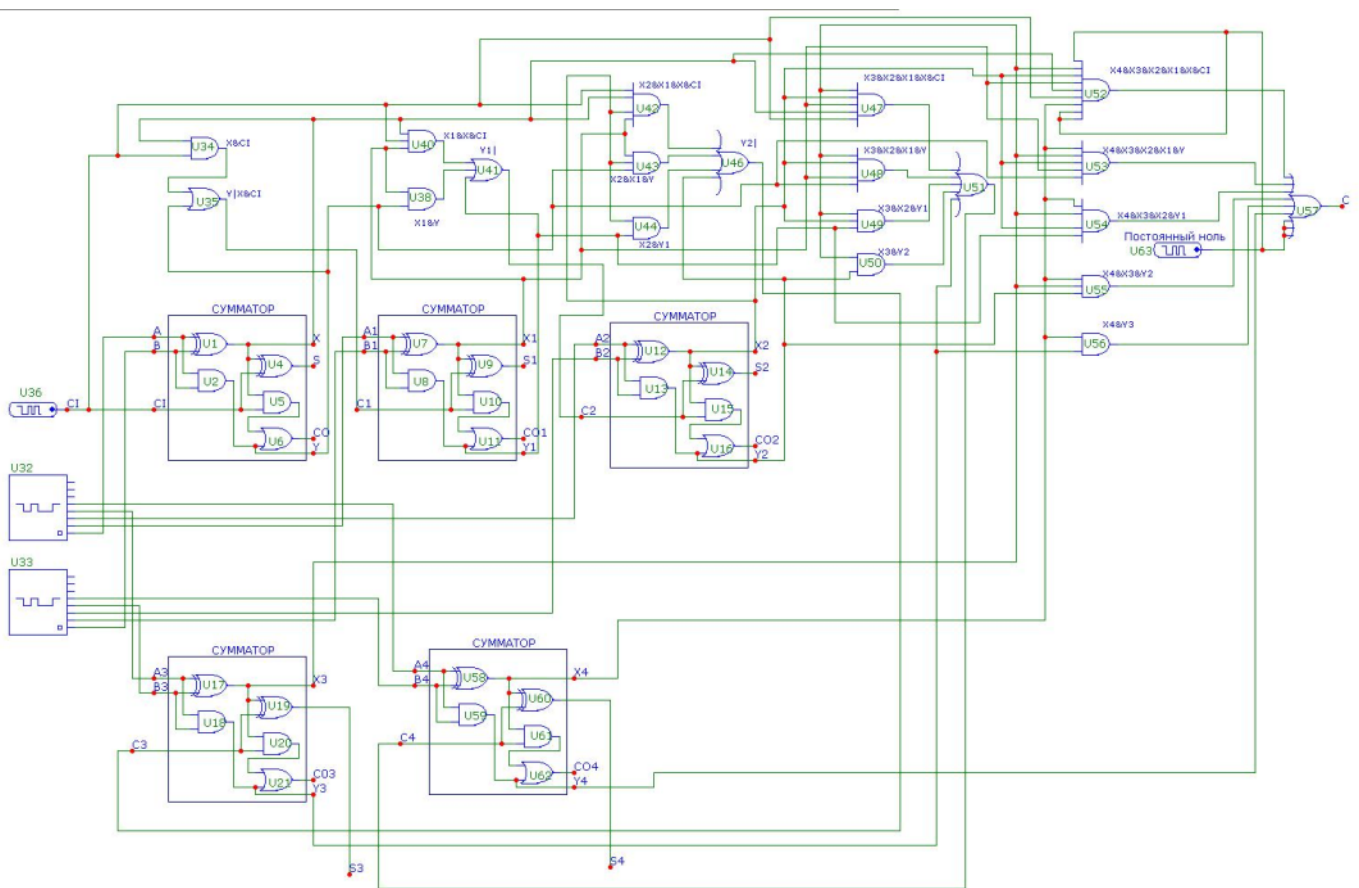


Рис.3. Схема сумматора параллельного переноса

Построение временных диаграмм сумматора параллельного переноса следует осуществлять, аналогично как в предыдущем пункте. При этом на диаграмме также следует обозначить время суммирования и время формирования переноса при задержке логических элементов равной 10 нс.

К п.4. Схема простейшего АЛУ (рис. 5) помимо сумматора включает дополнительные инвертирующие и переключающие элементы. Управление этими элементами осуществляется специальными сигналами U1, U2, U3, U4, которые поступают в АЛУ от устройства управления процессора. На рис. 6.

приведена схема простейшего четырехразрядного АЛУ выполняющего 7 элементарных операций.

Элементы "исключающее" ИЛИ U1-U6 служат для инвертирования разрядов слагаемых подаваемых на сумматоры. Управление инверсией того или иного слагаемого осуществляется управляющими сигналами U2 и U3.

Многоразрядное суммирование для простоты осуществим на сумматоре последовательного переноса, собранного на элементах X1-X3.

Мультиплексоры X4-X6 служат для коммутации к выходу АЛУ либо сигнала S, либо СО сумматоров. Выбор подключаемого сигнала осуществляется сигналом управления U4.

Сигнал управления U1 представляет собой сигнал переноса в младший разряд CO.

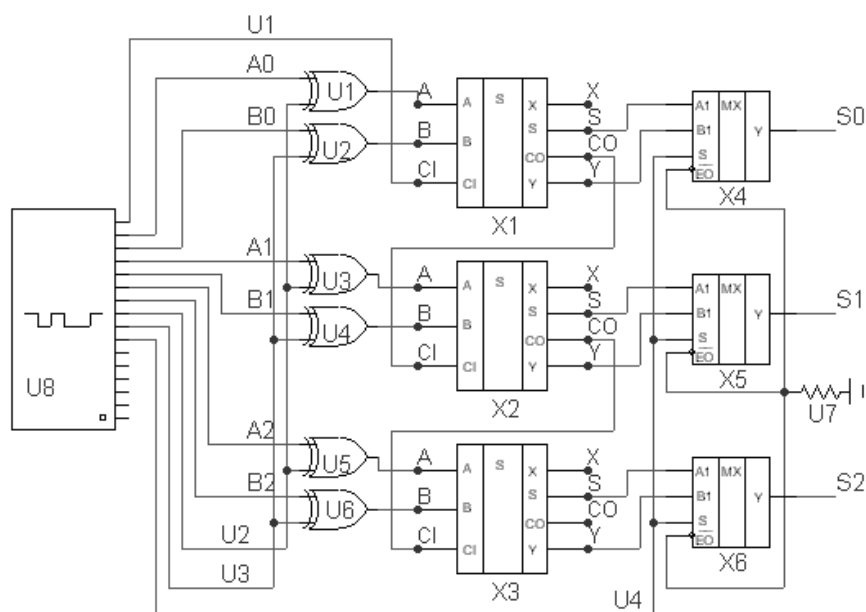


Рис. 4. Схема простейшего АЛУ

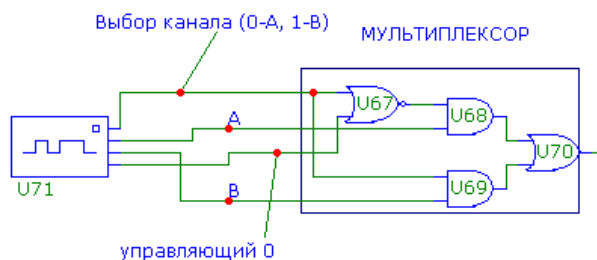


Рис. 5. Схема мультиплексора



В таблице 2 перечислены все основные выполняемые АЛУ функции, задачей студента является определить состояние управляющих входов U1-U4, для каждой из перечисленных функций.

Для моделирования работы АЛУ следует использовать 16-разрядный Stimulus Generator. С помощью него на входы АЛУ подаются требуемые слагаемые и сигналы управления. Слагаемые (или члены разности) выбираются по усмотрению студента. Зарисовать временные диаграммы и кратко пояснить используемый принцип вычитания.

В отчете приводится программа работы и после каждого пункта программы приводиться его выполнение.

Таблица - 2

Управляющие входы				Выполняемая функция
U1	U2	U3	U4	
				$S=A+B$
				$S=A+B+I$
				$S=A-B$
				$S=B-A$
				$S=A-B-I$
				$S=A \& B$
				$S=\overline{A} \& \overline{B} = \overline{A/B}$ (/ - "или")