Лабораторная работа №17
Шифрование с помощью управляемых операций

1. Цель работы

Освоить порядок моделирования криптосистемы, основанной на шифрование с помощью управляемых операций, в программе Multisim 11.0.2.

2. Общие сведения

Программа Multisim может быть успешно использована для имитации работы криптографических систем. В предыдущей работе был рассмотрен шифр гаммирования. Основная идея шифра гаммирования заключается в замене символов открытого текста на числа и суммировании их с псевдослучайными числами, которые называются «гаммой». При этом состав гаммы известен только доверенным лицам на передающей и приёмной сторонах.

Известны методы взлома этого шифра. Скомпрометировать шифр можно в случаях нештатного использования гаммы (некачественный состав гаммы, малая длина или повторное использование одной и той же гаммы для шифрования разных сообщений).

Ещё одним уязвимым элементом в аддитивном шифре является логическая операция Исключающее ИЛИ, которая используется для зашифрования и расшифрования.

Известно интересное свойство этой логической операции:

$$M \oplus G \oplus G = M .$$

Соотношение говорит о том, что наличие чётного числа одинаковых слагаемых, участвующих в операции Исключающее ИЛИ, уничтожает эти слагаемые. Таким образом, если определить период циклически повторяющейся гаммы и выполнить логическую операцию Исключающее ИЛИ над символами криптограммы с одинаковыми значениями гаммы (с одинаковыми фазами), то можно уничтожить гамму. В результате такого преобразования получаются данные, представляющие собой результат выполнения логической операции Исключающее ИЛИ над символами открытого текста:

$$R = C_i \oplus C_{i+T} = M_i \oplus G_i \oplus M_{i+T} \oplus G_{i+T} = M_i \oplus M_{i+T}$$

Это объясняется тем, что $G_i = G_{i+T}$, то есть элементы гаммы повторяются с периодом T и поэтому они одинаковые. Если гамма дважды использована для шифрования двух разных текстов, то задача криптоанализа становится ещё проще: достаточно выполнить операцию Исключающее ИЛИ над двумя криптограммами. Известен пример неверного использования метода гаммирования в операционной системе Windows 95. Одна и та же гамма применялась несколько раз для шифрования данных в файлах PWL (эти файлы

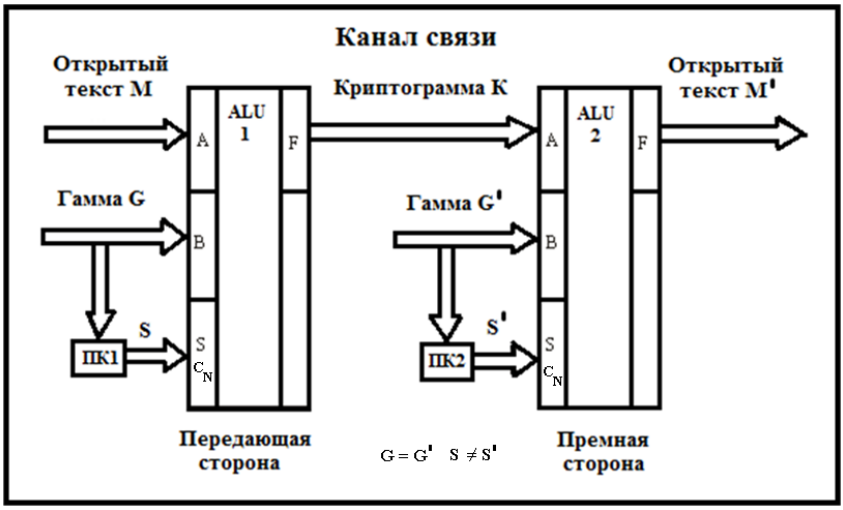
хранят логины и пароли).

Величину R можно назвать **разностью открытых текстов** (сообщений). Разность R может быть подвержена успешному криптоанализу путём учёта статистических закономерностей открытых текстов или использования известных из других источников их особенностей.

Таким образом, в аддитивном методе шифрования из-за симметричности (обратимости) логической операции Исключающее ИЛИ и нештатного использования гаммы существует возможность произвести криптоанализ и восстановить открытый текст даже без знания гаммы.

Повысить криптостойкость аддитивного шифра можно за счёт использования управляемых операций шифрования [3, 4]. Основная идея данной криптосистемы состоит в использовании в течение одного сеанса связи не одной, а нескольких различных шифрующих операций. В этой криптосистеме вместе с изменением значения гаммы варьируются операции преобразования, выполняемые над открытым текстом (на передаче) и над криптограммой (на приёме). Причём на передаче и приёме операции зашифрования и расшифрования должны синхронно чередоваться. Например, если на передаче осуществляется арифметическое сложение символа открытого текста с элементом гаммы, то на приёме нужно вычесть гамму из полученной криптограммы. Синхронизация выполняемых операций должна осуществляться под управлением гаммы, которая одновременно определяет и вид выполняемой операции, и сама участвует в этих операциях.

На рисунке показана структурная схема криптографической системы с управляемыми операциями шифрования.



Имитация передающей и приёмной сторон криптосистемы осуществляется с помощью двух арифметико-логических устройств. Четыре разряда открытого текста *M* подаётся на вход A первого арифметико-логического устройства (АЛУ). Четырехбитная гамма *G* подаётся на входы *B* каждого АЛУ. Вид выполняемой операции на передающей стороне задаётся с помощью преобразователя кода ПК1. Управляющие сигналы S на приёмной стороне формируются с помощью преобразователя кода ПК2. Сигналы с выходов преобразователей кодов подаются на управляющие шины АЛУ и шину переноса из старшего разряда C_N. Именно эти сигналы определяют вид выполняемых АЛУ операций.

Криптограмма *K* формируется на выходе F первого АЛУ. Расшифрование криптограммы происходит на приёмной стороне с помощью второго АЛУ. Выполняемые операции синхронно изменяются под управлением гаммы. Принятый открытый текст *M`* появляется на выходе F второго АЛУ.

В качестве шифрующих преобразований можно использовать различные логические и арифметические операции, а также математические функции и их комбинации [5]. Некоторые из них перечислены в таблице 2.1, в которой приняты такие обозначения:

M - открытый текст (сообщение); *G* - гамма; *K* - криптограмма; $\oplus$ - логическая операция Исключающее ИЛИ (неравнозначность); $\leftrightarrow$ - логическая операция равнозначность; «+» - арифметическая операция сложение; «–» - арифметическая операция вычитание; черта над переменными обозначает операцию инверсии.

Первые 10 операций в таблице 2.1 предполагают работу ЭВМ с целыми числами. Остальные операции предназначены для работы с вещественными числами.

Задачей преобразователей кодов ПК1 и ПК2 является синхронное изменение управляющих сигналов на передающей и приёмной сторонах. Естественно, что конструкции преобразователей кодов ПК1 и ПК2 должны быть разными, так как при одинаковых входных воздействиях (гамма *G*) преобразователи кодов должны формировать разные выходные (управляющие) сигналы S, S` и сигналы переносов C_N.

Преобразователи кодов можно синтезировать различными способами: графически (с помощью карт Карно и диаграмм Вейча), аналитически (методы Квайна, Мак-Класки, неопределённых коэффициентов) и с помощью графических символов, интерпретирующих булевы функции.

Перечисленные способы синтеза комбинационных цифровых устройств трудоёмки и имеют ограничения на их использование при числе переменных более 5…6. При разработке модели данной криптосистемы преобразователи кодов целесообразно синтезировать с помощью блока Logic Converter (логический конвертор), который входит в систему моделирования радиоэлектронных устройств Multisim.

Таблица 2.1

	Операции на **передающей** стороне	Операции на **приёмной** стороне
1.	Неравнозначность $K = M \oplus G$	Неравнозначность $M` = K \oplus G$
2.	Равнозначность $K = \overline{M \oplus G} = M \leftrightarrow G$	Равнозначность $M` = \overline{K \oplus G} = K \leftrightarrow G$
3.	Сложение $K = M + G$	Вычитание $M` = K - G$
4.	Вычитание $K = M - G$	Сложение $M` = K + G$
5.	Вычитание $K = G - M$	Вычитание $M` = G - K$
6.	Инверсия от суммы $K = \overline{M + G}$	Комбинированная разность $M` = \overline{K} - G$
7.	Инверсия от разности $K = \overline{M - G}$	Комбинированная сумма $M` = \overline{K} + G$
8.	Инверсия от разности $K = \overline{G - M}$	Комбинированная разность $M` = G - \overline{K}$
9.	Комбинированная сумма $K = \overline{M} + \overline{G}$	Комбинированная разность $M` = \overline{K - \overline{G}}$
10.	Комбинированная разность $K = M - \overline{G}$	Комбинированная сумма $M` = K + \overline{G}$
11.	Умножение $K = M \cdot G$	Деление $M = K / G$
12.	Деление $K = M / G$	Умножение $M = K \cdot G$
13.	Деление $K = G / M$	Деление $M = G / K$
14.	Функциональные $K = f(M, G)$	Функциональные $M = f^{-1}(K, G)$
15.	Алгебраические $K = M^n \pm G^s$	Алгебраические $M = \sqrt[n]{K \mp G^s}$

Логический конвертор позволяет создавать преобразователи кодов с числом аргументов $n \le 8$. Для получения логических выражений, описываю-

щих работу ПК, достаточно в конвертор ввести таблицу истинности, которая описывает работу преобразователя кода. Полученные математические выражения затем можно использовать для построения принципиальной схемы ПК.

Рассмотрим подробнее порядок выбора логических и арифметических операций, которые можно использовать в криптосистеме.

Безусловно, для шифрования нужно использовать многократно проверенную на практике операцию Исключающее ИЛИ. Аналогичными положительными свойствами обладает операция “Равнозначность”, которая является инверсной по отношению к операции Исключающее ИЛИ.

В виду того, что логические операции $\overline{\mathrm{M}} \leftrightarrow \mathrm{G} = \mathrm{M} \leftrightarrow \overline{\mathrm{G}}$ эквивалентны операции неравнозначности $\mathrm{M} \oplus \mathrm{G}$, использовать все три операции при шифровании не имеет смысла, так как криптограммы при шифровании будут одинаковыми. Аналогично операции $\overline{\mathrm{M}} \oplus \mathrm{G} = \mathrm{M} \oplus \overline{\mathrm{G}}$ сводятся к операции равнозначности $\mathrm{M} \leftrightarrow \mathrm{G}$. Таким образом, из рассмотренных шести операций следует использовать только две: равнозначность и неравнозначность.

Для арифметических операций в дополнительном коде справедливы соотношения:

$$\overline{M} - G = \overline{G} - M = \overline{M + G}$$
$$\overline{M} + G = \overline{G} - \overline{M} = \overline{M - G}$$
$$\overline{G - M} = \overline{\overline{M} - \overline{G}} = M + \overline{G}$$
$$G - \overline{M} = \overline{\overline{M} + \overline{G}} = M - \overline{G}$$
$$\overline{M} - \overline{G} = G - M$$

Использование операций, перечисленных в одной строке, даст одинаковые значения криптограммы при одинаковых значениях гаммы и открытого текста. Из четырнадцати указанных операций целесообразно оставить только шесть, например,

$$\overline{M + G},\ \overline{M - G},\ M + \overline{G},\ M - \overline{G}, M - G \text{ и } G - M.$$

3. Задания на выполнение лабораторной работы

3.1. Задание 1. Синтез преобразователя кода

Используя таблицу 3.1.1, разработать два преобразователя кода (по одному для передающей и приёмной стороны). Преобразователи кодов нужно синтезировать с помощью блока Logic Converter (логический конвертор).

Таблица 3.1.1

Варианты	Значения гаммы G			
	$M \oplus G$	$\overline{M \oplus G}$	$M - G$	$M + G$
1	0,5,6,7	1,3,11	2,8,12,15	4,9,10,13,14
2	2,3,7,11	8,12,14,15	0,1,5,9,13	4,6,10
3	0,1,4,5	2,3,12,14,15	6,8,10	7,9,11,13
4	0,13,14,15	4,6,8,10,12	1,3,5	2,7,9,11
5	1,5,9,13	3,7,11,15	2,6,10,14	0,4,8,12
6	0,5,10,15	3,6,9,12	4,8,7,11	1,2,13,14
7	0,4,8,12	1,5,9,13	2,6,10,14	3,7,11,15
8	2,6,10,14	0,4,8,12	3,7,11,15	1,5,9,13
9	3,7,11,15	2,6,10,14	1,5,9,13	0,4,8,12
10	4,5,8,9,	2,3,12,13	0,1,6,7	10,11,14,15
11	13,15,3,7	2,6,9,12	0,4,8,11	1,5,10,14
12	2,3,6,7	10,11,14,15	4,5,8,9	0,1,12,13
13	2,6,8,12	3,7,11,15	0,4,10,14	1,5,9,13
14	5,7,10,13	4,6,12,15	0,2,8,11	1,3,9,14
15	0,4,9,13	1,5,8,12	3,7,10,14	2,6,11,15
16	3,7,8,12	0,4,10,14	2,6,11,15	1,5,9,13
17	0,1,2,3	4,5,6,7	8,9,10,11	12,13,14,15

Таблица 3.1.1 показывает, какую операцию должна использовать криптосистема для указанного десятичного значения гаммы. Например, для варианта 17, если гамма равна 0, то выполняемая операция будет $M \oplus G$. Если значение гаммы равно 11, то операция на передающей стороне будет $M - G$ и т.д.

3.2. Задание 2. Разработка принципиальной схемы криптосистемы и моделирование её работы

Криптосистема должна работать с использованием четырёх операций: Исключающее ИЛИ, равнозначность, сложение и вычитание (текст минус гамма). Эти операции должны сменять друг друга в зависимости от значений гаммы. Для реализации этого при составлении принципиальной схемы криптосистемы следует использовать разработанные в предыдущем задании преобразователи кода. Именно преобразователи кодов выполняют управление работой АЛУ (изменение шифрующих и дешифрующих операций).

Составленную принципиальную схему криптосистемы следует использовать для моделирования её работы. В процессе моделирования необходимо проверить выполнение шестнадцати логических и арифметических операций. Значения операндов открытого текста в зависимости от значения гаммы и номера варианта нужно выбрать из таблицы.

Таблица 3.2.1

Гамма (G) / Варианты	Открытый текст (M)							
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	1	2	3	4	5	6	7	8
2	2	3	4	5	6	7	8	9
3	3	4	5	6	7	8	9	10
4	4	5	6	7	8	9	10	11
5	5	6	7	8	9	10	11	12
6	6	7	8	9	10	11	12	13
7	7	8	9	10	11	12	13	14
8	8	9	10	11	12	13	14	15
9	9	10	11	12	13	14	15	0
10	10	11	12	13	14	15	0	1
11	11	12	13	14	15	0	1	2
12	12	13	14	15	0	1	2	3
13	13	14	15	0	1	2	3	4
14	14	15	0	1	2	3	4	5
15	15	0	1	2	3	4	5	6
16	0	1	2	3	4	5	6	7
17	3	0	1	2	6	7	4	5

Продолжение таблицы 3.2.1

Гамма (G) / Варианты	Открытый текст (M)							
	8	**9**	**10**	**11**	**12**	**13**	**14**	**15**
1	9	10	11	12	13	14	15	0
2	10	11	12	13	14	15	0	1
3	11	12	13	14	15	0	1	2
4	12	13	14	15	0	1	2	3
5	13	14	15	0	1	2	3	4
6	14	15	0	1	2	3	4	5
7	15	0	1	2	3	4	5	6
8	0	1	2	3	4	5	6	7
9	1	2	3	4	5	6	7	8
10	2	3	4	5	6	7	8	9
11	3	4	5	6	7	8	9	10
12	4	5	6	7	8	9	10	11
13	5	6	7	8	9	10	11	12
14	6	7	8	9	10	11	12	13
15	7	8	9	10	11	12	13	14
16	8	9	10	11	12	13	14	15
17	10	11	8	9	14	15	12	13

Таблицу 3.2.1 нужно трактовать так. Значения гаммы G для всех вариантов одинаковые 0…15 (верхняя строка чисел). Открытый текст M имитируется числами (для каждого варианта своя последовательность). Например, для варианта 17 эта последовательность такова 3 - 0 - 1- 2…12 - 13.

Результаты моделирования следует сопоставить с результатами ручных расчетов и занести в таблицу. Ниже приведена форма этой таблицы.

Таблица 3.1.3

№ п/п	Значение гаммы (G)	Значение открытого текста (M). Вариант 17	Значение криптограммы (K). Результаты моделирования	Значение криптограммы (K). Результаты ручного расчета
1	0	3		
2	1	0		
…	…	…		
16	15	13		

4. Порядок выполнения лабораторной работы

4.1. Методические указания к заданию 3.1

Открытый текст, гамма, криптограмма в криптосистеме с управляемыми операциями представлены четырехразрядными двоичными операндами (тетрадами), а управляющие сигналы формируются с помощью пяти двоичных разрядов. Ещё один разряд потребуется для формирования сигнала переноса. Таким образом, преобразователь кода должен содержать четыре входа (на них подаётся гамма) и шесть выходов.

Для синтеза преобразователя кода на **передающей стороне** нужно составить таблицу, которая формируется на основании данных из таблицы 3.1.1. Таблица 4.1.1 составлена для варианта 17

Таблица 4.1.1

№ п/п	Гамма $B_3B_2B_1B_0$ (ABCD)	Операция	Управляющие сигналы $S_3S_2S_1S_0$	Mod	C_N
0	0 0 0 0	$M \oplus G$	0 1 1 0	1	x
1	0 0 0 1		0 1 1 0		
2	0 0 1 0		0 1 1 0		
3	0 0 1 1		0 1 1 0		
4	0 1 0 0	$\overline{M \oplus G}$	1 0 0 1	1	x
5	0 1 0 1		1 0 0 1		
6	0 1 1 0		1 0 0 1		
7	0 1 1 1		1 0 0 1		
8	1 0 0 0	$M - G$	0 1 1 0	0	0
9	1 0 0 1		0 1 1 0		
10	1 0 1 0		0 1 1 0		
11	1 0 1 1		0 1 1 0		
12	1 1 0 0	$M + G$	1 0 0 1	0	1
13	1 1 0 1		1 0 0 1		
14	1 1 1 0		1 0 0 1		
15	1 1 1 1		1 0 0 1		

В колонке «Гамма» одновременно указаны разряды АЛУ ($B_3B_2B_1B_0$),

на которые подаётся гамма, и обозначения этих же разрядов в конверторе Logic Converter (ABCD).

Первые две конки таблицы будут одинаковыми для всех вариантов.

Колонка «Операция» заполняется следующим образом (на примере варианта № 17). В соответствии с таблицей 3.1.1 для гаммы, равной значениям «0», «1», «2», «3», должна выполнятся операция $M \oplus G$, поэтому в строках 0-3 четырежды указана данная операция. Другие строки таблицы заполняются аналогично. Понятно, что в других вариантах последовательность операций будет иной.

Колонка «Управляющие сигналы» заполняется с учётом информации, содержащейся в колонке «Операция». Соответствие управляющих сигналов $S_3S_2S_1S_0$ и выполняемых АЛУ операций следующее:

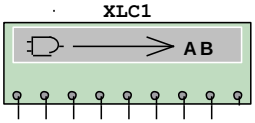
$$\text{«}M \oplus G\text{»} - \text{«0 1 1 0»},\ \text{«}\overline{M \oplus G}\text{»} - \text{«1 0 0 1»},$$

$$\text{«}M - G\text{»} - \text{«0 1 1 0»},\ \text{«}M + G\text{»} - \text{«1 0 0 1»}$$

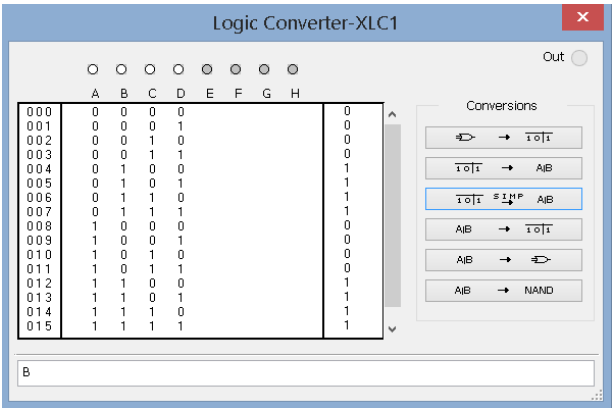
Входы $S_3S_2S_1S_0$ и Mod предназначены для формирования управляющих сигналов, которые позволяют выбрать одну из тридцати двух возможных операций АЛУ. Напомним, что в данной работе используются лишь четыре операции: две логические операции и две арифметические.

Столбец «Mod» определяет, какую операцию выполняет АЛУ: логическую (значение равно 1) или арифметическую (значение равно 0). Сигнал C_N определяет величину переноса. Для логических операций безразлично, какое значение принимает C_N, поэтому сигнал обозначен символом «x». Для арифметического вычитания сигнал C_N равен «0», а для арифметического сложения – «1».

На основании данных из таблицы 4.1.1 следует сформировать логические выражения, которые будут использованы для синтеза преобразователя кода. Составить логические выражения проще всего с помощью Logic Converter (логический конвертор).



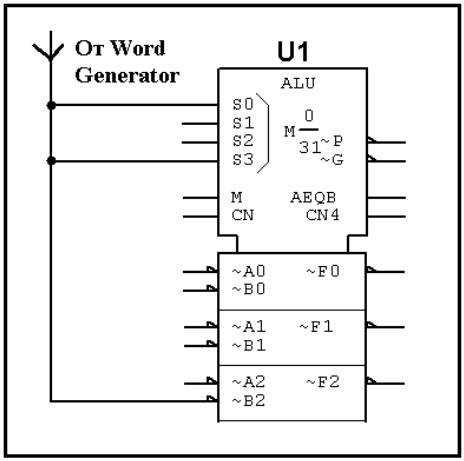
Для получения математических выражений, описывающих работу ПК, достаточно в конвертор ввести данные из таблицы истинности преобразователя кода. В качестве примера получим выражение для S_3 (вариант 17). Логические выражения формируются при нажатии кнопки **SIMP**, находящейся в поле **Conversions**.



Как видно из приведённого рисунка, вся введённая в конвертор таблица истинности описывается одним символом *B*. Анализируя таблицу 4.1.1, несложно заметить, что сигналы на шинах S_3 и S_0 должны совпадать, поэтому эти два входа АЛУ должны быть соединены между собой.

Учитывая последовательность ввода операндов и принятые обозначения (см. табл. 4.1.1), можно записать: $S_3 = S_0 = B_2$.

Фрагмент схемы ПК, который реализует полученное выражение показан на рисунке.



Напомним, что символ *A* в окне Logic Converter соответствует разряду гаммы B_3, символ $B - B_2$, $C - B_1$, $D - B_0$.

Описанные действия по определению сигнала на одном выходе ПК

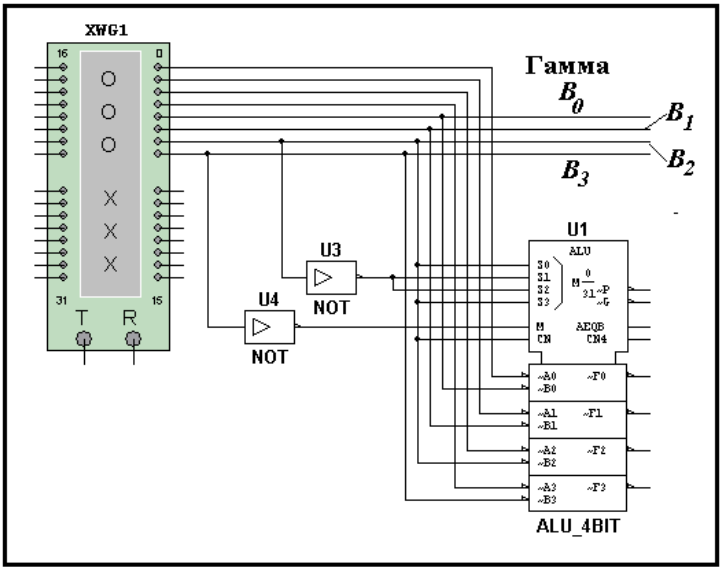
следует повторить для всех выходов ПК. В результате должно быть получено шесть логических выражений, часть из которых могут быть одинаковыми.

Например, анализируя таблицу 4.1.1, легко заметить, что сигналы $S_2 = S_1$. Это означает, что эти шины должны быть соединены между собой.

Ниже приведены результаты формирования логических выражений для остальных четырёх выходных сигналов первого ПК:

$$S_2 = S_1 = \overline{B}_2,\ M = \overline{B}_3,\ C_N = B_2.$$

Эти выражения следует использовать для построения схемы преобразователя кода на передаче. Вся схема будет состоять из двух инверторов для сигналов B_1 и B_3. Ещё один выходной сигнал ПК берётся непосредственно с шины B_2. На рисунке показана шифрующая часть криптосистемы (передающая часть).



Затем следует выполнить синтез ПК **на приёмной стороне**. Для этого, используя данные таблицы 3.1.1, нужно составить новую таблицу истинности. При этом следует помнить, что логические операции на приёме и передаче должны совпадать, а арифметические операции на приёме должны поменяться на противоположные, то есть, если на передаче использовалось арифметическое сложение, то на приёме нужно использовать арифметическое вычитание, и наоборот. В итоге таблица истинности примет следующий вид:

Таблица 4.1.2

№ п/п	Гамма $B_3B_2B_1B_0$ (ABCD)	Операция	Управляющие сигналы $S_3S_2S_1S_0$	Mod	C_N
0	0 0 0 0	$M \oplus G$	0 1 1 0	1	x
1	0 0 0 1		0 1 1 0		
2	0 0 1 0		0 1 1 0		
3	0 0 1 1		0 1 1 0		
4	0 1 0 0	$\overline{M \oplus G}$	1 0 0 1	1	x
5	0 1 0 1		1 0 0 1		
6	0 1 1 0		1 0 0 1		
7	0 1 1 1		1 0 0 1		
8	1 0 0 0	$M + G$	1 0 0 1	0	1
9	1 0 0 1		1 0 0 1		
10	1 0 1 0		1 0 0 1		
11	1 0 1 1		1 0 0 1		
12	1 1 0 0	$M - G$	0 1 1 0	0	0
13	1 1 0 1		0 1 1 0		
14	1 1 1 0		0 1 1 0		
15	1 1 1 1		0 1 1 0		

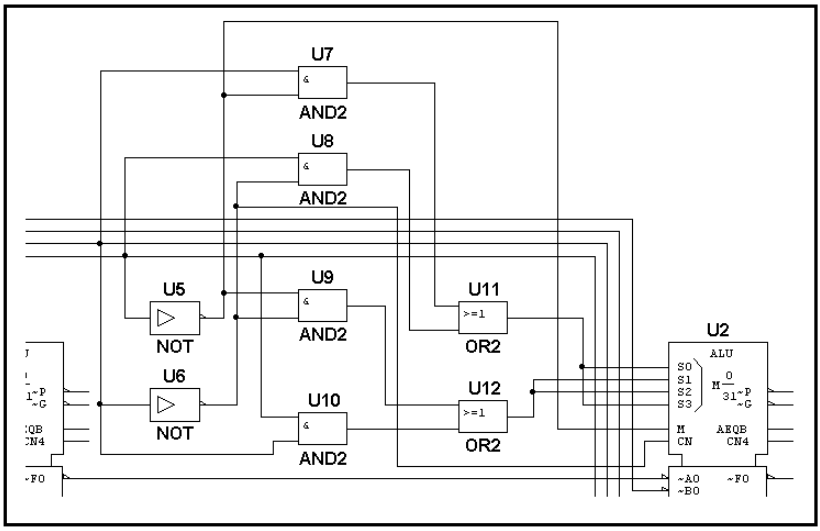
Используя логический конвертор и данные из таблицы, несложно получить выражения, которые описывают ПК на приёмной стороне:

$$S_3 = S_0 = (\overline{B}_3 \wedge B_2) \vee (B_3 \wedge \overline{B}_2),$$

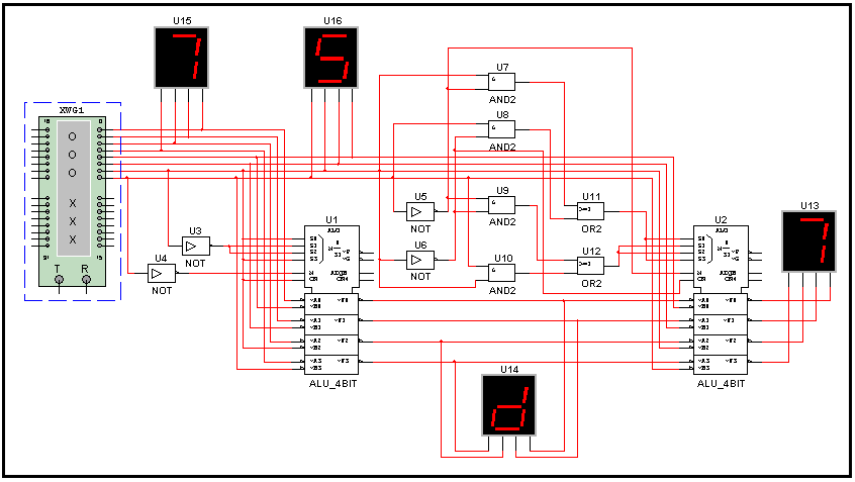
$$S_2 = S_1 = (\overline{B}_3 \wedge \overline{B}_2) \vee (B_3 \wedge B_2),$$

$$M = \overline{B}_3,\ C_N = \overline{B}_2.$$

На основе этих выражений формируется схема ПК на приёмной стороне. Следующий рисунок показывает схему второго ПК.



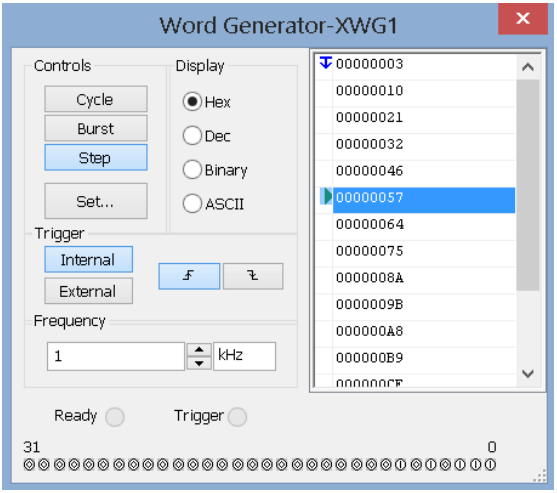
Полная принципиальная схема криптосистемы имеет вид:



Четырехбитный открытый текст подаётся на вход A первого арифметико-логического устройства (U1). Гамма подаётся на вход B. Вид выполняемой операции задаётся с помощью преобразователя кода (U3, U4). Криптограмма формируется на выходе F первого АЛУ (U1). Дешифрация криптограммы осуществляться на приёмной стороне с помощью второго АЛУ (U2). Вид выполняемой операции синхронно изменяется под управлением гаммы.

Принятый открытый текст появлялся на выходе F второго АЛУ (U2).

Исходный текст, принятый текст, гамма и криптограмма отображаются с помощью четырёх индикаторов соответственно U15, U13, U16, U14. Значения гаммы и передаваемый текст формируются с помощью генератора слов XWG1 (Word Generator). На рисунке показан генератор слов, в буфере которого содержатся данные для варианта 17.



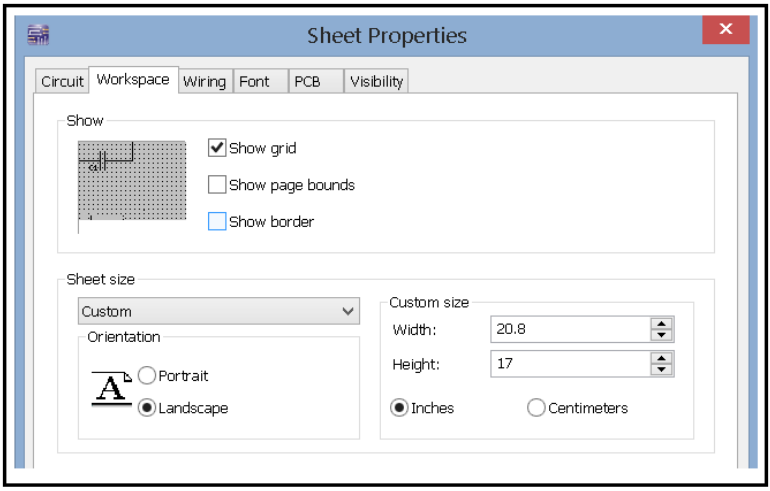
Как видно из рисунка, выполняется 6-я по счету операция. В соответствии с таблицей 4.1.2 в этот момент времени выполняется шифрующая операция «Равнозначность». Операция выполняется над числами 5 и 7 и результат равен шестнадцатеричному числу D. На приёмной стороне число подвергается дешифрации и в результате получается число 7 (открытый текст). Именно эти числа показаны на принципиальной схеме криптосистемы.

На последнем этапе выполнения лабораторной работы следует проверить правильность работы созданной криптосистемы. Для этого в окне свойств генератора слов XWG1 нужно выставить заданные значения гаммы и открытого текста, как показано на рисунке, и нажатием на клавишу **Step**, проверить все операции. Для удобства флажок таблицы исчисления должен стоять на отметке **Hex**.

Результаты ручных расчётов и моделирования криптосистемы следует внести в таблицу 3.1.3.

В рассмотренном примере синтеза криптосистемы принципиальные схемы преобразователей кодов достаточно просты. При существенном усложнении конструкции возникает проблема из-за высокой плотности размещения микросхем на отведённой площади листа. Увеличить пространство рабочего стола можно с помощью диалогового окна, показанного на сле-

дующем рисунке. Вызывается диалоговое окно с помощью контекстного меню и выбора опции Properties (Свойства).



Как и всякая имитация, рассмотренная модель не полностью соответствует реальной криптографической системе. Например, при моделировании предполагается, что соединение между передающей и приёмной сторонами происходит по четырём проводам. В реальной криптосистеме связь должна осуществляться по двухпроводной линии.

Кроме того, при моделировании считается, что операнды, циркулирующие в криптосистеме, являются четырёхразрядными целыми числами. Диапазоны изменения чисел составляли $0 \le M \le 15$ и $0 \le G \le 15$. Заметим, что в реальной криптосистеме при формировании криптограммы возможно использование не только целых, но и вещественных чисел.

5. Требования к отчёту

Отчёт подготавливается в электронном виде. Он должен содержать постановки задач, скриншоты, которые показывают порядок решения задач, схемы, таблицы с результатами моделирования, необходимые комментарии и анализ полученных результатов.

6. Контрольные вопросы

6.1. В чём состоит основная идея шифрования с помощью управляемых операций?
6.2. Выполните операцию вычитания в дополнительном коде.
6.3. Запишите закон де Моргана.
6.4. Перечислите возможности прибора Logic Converter –XLC1.
6.5. Составьте таблицу истинности логического элемента ИЛИ-НЕ
6.6. Составьте таблицу истинности логического элемента И-НЕ.
6.7. Запишите тождества алгебры логики.
6.8. Запишите законы алгебры логики.
6.9. Как представить отрицательное число в дополнительном коде?
6.10. Чем отличаются логические и арифметические операции?
6.11. Для чего используются в АЛУ выводы M и C_N?

7. Список литературы

1. Алексеев А.П. Информатика 2015 [Текст]: учеб. пособие/ Алексеев А.П. – М: СОЛОН-Пресс, 2015. – 400 с. ISBN 978-5-91359-158-6.
2. Алексеев А.П. Информатика для криптоаналитиков: учебное пособие/ Алексеев А.П. – Самара: ИУНЛ ПГУТИ, 2015. – 376 с. ISBN 978-5-904029-53-1.
3. Алексеев А. П., Моделирование криптосистемы с управляемыми операциями с помощью MULTISIM/ Алексеев А. П., Жеренов Ю.В., Орлов В. В. // Инфокоммуникационные технологии, том 7, № 4, 2009. Стр. 78-82.
4. Криптография: скоростные шифры/ Молдовян А.А., Молдовян Н.А., Гуц Н.Д., Изотов Б.В. - СПб: БХВ – Петербург, 2002. - 496 с.
5. Алексеев А.П., Тамаров Р.В. Критерий отбора функций для многоалфавитного шифра с большим числом математических преобразований. Доклад на XIV международной научно-технической конференции «Проблемы техники и технологий телекоммуникаций» ПТиТТ-2013. –стр. 240 - 241.