

Лабораторная работа №3

Разработка преобразователей формата представления числа: двоичный – двоично-десятичный, двоично-десятичный – двоичный

Цель работы: Изучить алгоритмы преобразования форматов представления числа разработать Verilog-описания двух преобразователей:

- 1) из двоичного в двоично-десятичный;
- 2) из двоично-десятичного в двоичный.

Также необходимо выполнить моделирование работы разработанных преобразователей, и выполнить конфигурацию ПЛИС для проверки на отладочном стенде с использованием стандартных устройств ввода-вывода информации, проверив работоспособность разработанных преобразователей на учебном стенде.

После этого нужно выполнить индивидуальное задание.

Продолжительность работы: 4 академических часа.

Оборудование, приборы, инструментарий: учебный стенд Altera DE1, САПР Altera Quartus II.

Теоретическая часть.

Двоично-десятичный код (*BCD – Binary Code Decimal*) - это форма записи целых чисел, когда каждый десятичный разряд числа записывается в виде его четырехбитного двоичного кода (тетрада или полубайт). Например, десятичное число 46_{10} будет записано в двоично-десятичном коде как $0100\ 0110_{BCD}$, где $0100_2 = 4_{10}$, а $0110_2 = 6_{10}$ (в то время как в двоичном коде $46_{10} = 0010\ 1110_2$).

В двоично-десятичном формате каждый разряд десятичного числа представляется в виде четырехбитной двоичной тетрады (полубайта), как показано в Таблице 1.

Таблица 1. Примеры представления десятичного числа в двоично-десятичном формате.

Десятичный формат	Двоично-десятичный формат	Двоичный формат
4	0100	0100
15	0001 0101	1111
89	1000 1001	0101 1001
132	0001 0011 0010	1000 0100

Каждый разряд десятичного числа D_i не может превышать значение 9, поэтому двоичные коды $1010_2 - 1111_2$ ($10_{10} - 15_{10}$) в записи тетрад двоично-десятичного формата не используются. Это накладывает дополнительные условия на операции сложения и вычитания, требуя ввода корректирующего значения (см. Список литературы).

Обобщим достоинства и недостатки двоично-десятичной записи чисел.

Достоинства:

- Упрощён вывод чисел на индикацию (например, на 7-сегментный дисплей) – вместо последовательного деления на 10 требуется просто вывести на индикацию каждый полубайт. Аналогичным образом упрощён ввод данных с цифровой клавиатуры.
- Для дробных чисел (как с фиксированной, так и с плавающей запятой) при переводе в десятичный формат и наоборот не теряется точность.
- Упрощены умножение и деление на 10, а также округление.

По этим причинам двоично-десятичный формат применяется в калькуляторах – калькулятор при выполнении простейших арифметических операций должен выводить в точности такой же результат, какой подсчитает человек на бумаге.

Недостатки:

- Требуется больше памяти.
- Усложнены арифметические операции, так как в двоично-десятичном формате используются только 10 возможных комбинаций 4-х битового поля вместо 16, существуют запрещённые комбинации битов: 1010(1010), 1011(1110), 1100(1210), 1101(1310), 1110(1410) и 1111(1510).

Методические указания.

Необходимо разработать Verilog-описание преобразователей форматов из двоичного в двоично-десятичный (модуль bin2dec) и из двоично-десятичного в двоичный (модуль dec2bin).

Часть 1

Исходное двоичное число должно быть переведено в двоично-десятичный формат. Каждая тетрада в двоично-десятичном коде должна отображаться на 7-сегментном индикаторе, дешифратор которого был разработан при выполнении предыдущей лабораторной работы.

На первом этапе необходимо получить число $\mathbf{m} = \{\mathbf{m}_3, \mathbf{m}_2, \mathbf{m}_1, \mathbf{m}_0\}$ в двоично-десятичном формате из двоичного числа $\mathbf{v} = \{\mathbf{v}_3, \mathbf{v}_2, \mathbf{v}_1, \mathbf{v}_0\}$. Число \mathbf{v} предлагается вводить с помощью переключателей \mathbf{SW} , изменяя значения переключателей на логические «0» или «1» для задания необходимого значения. Четырехразрядное значение \mathbf{SW}_{3-0} , может кодировать число от $\mathbf{4'h0}$ до $\mathbf{4'hF}$. Преобразователь должен отображать на дешифратор индикатора цифры в диапазоне 0 до 9, а при вводе числа из диапазона от 10 до 15 сигнализировать об ошибке («ноль»). В соответствии с этими соображениями составим таблицу истинности преобразователя (таблица 2).

Таблица 2. Таблица истинности схемы разрабатываемого преобразователя

$\{\mathbf{v}_3 \mathbf{v}_2 \mathbf{v}_1 \mathbf{v}_0\}$	$\{\mathbf{m}_3 \mathbf{m}_2 \mathbf{m}_1 \mathbf{m}_0\}$
0000	0000
0001	0001
0010	0010
0011	0011
0100	0100
0101	0101
0110	0110
0111	0111
1000	1000
1001	1001
1010	0000
1011	0000
1100	0000
1101	0000
1110	0000
1111	0000

В соответствии с таблицей истинности предлагается схема преобразователя (см. рисунок 1). Компаратор **СМР** сравнивает значение, заданное с помощью переключателей со значением 9, если оно больше, то сигнал **z** выставляется в логическую «1».

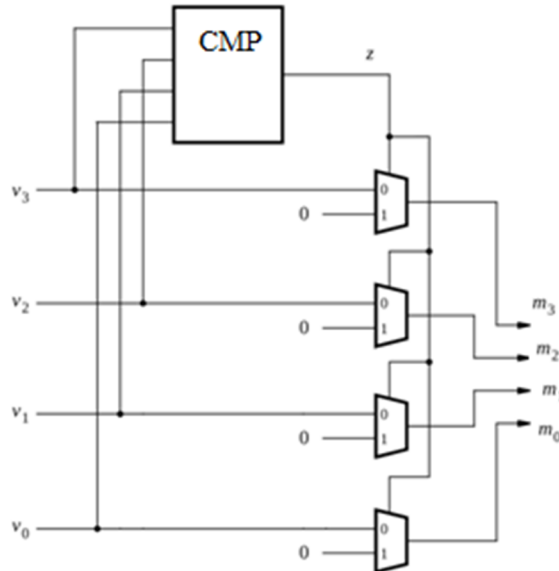


Рисунок 1 – Схема преобразователя bin2dec

Модуль преобразователя должен описывать компаратор, формирующий сигнал **z**, который затем на основании входных значений **v** определяет выходные сигналы **m**.

```
module bin2dec (v, m);
input [3:0] v;
output [3:0] m;
wire z;
...
endmodule
```

Скомпилируйте проект, промоделируйте работу модуля bin2dec, проверив правильность его работы, используя графические диаграммы, сверяя её с таблицей 2.

Часть 2

После выполнения части 1 у нас есть схема, которая передаёт на выход введённое значение, если оно меньше 10, и «0», если оно больше или равно 10. Как теперь модифицировать схему, позволяющую корректно отображать значение, большее 9, на 7-сегментных индикаторах?

Для решения этой задачи предлагается схема, изображённая на рисунке 2, которая позволяет преобразовать 4-разрядное двоичное число в двоично-десятичный формат и вывести получившиеся значения на два 7-сегментных индикатора. В этой схеме схема из части 1 дополнена следующими элементами:

- 7-сегментные индикаторы **HEX0** и **HEX1**;
- дешифратор **DEC**, предназначенный для отображения чисел от 0 до 9 на 7-сегментном индикаторе **HEX0**;
- **схема А**, предназначенная для установки значения, передаваемое на декодер при $z = 1$ (введённое число больше 9);
- **схема Б**, предназначенная для установки значения «1» на 7-сегментный индикатор **HEX1**, предназначенный для отображения десятков десятичного числа.

Составим таблицу истинности для **схемы А** для входного значения **v** и выходных значений **z**, **m** и **d**:

Таблица 2 - Таблица истинности двоично-десятичного преобразователя.

$v_3v_2v_1v_0$	z	$m_3m_2m_1m_0$	d₁	d₀
0000	0	0000		0
0001	0	0001		1
0010	0	0010		2
0011	0	0011		3
0100	0	0100		4
0101	0	0101		5
0110	0	0110		6
0111	0	0111		7
1000	0	1000		8
1001	0	1001		9
1010	1	0000	1	0

$v_3v_2v_1v_0$	z	$m_3m_2m_1m_0$	d_1	d_0
1011	1	0001	1	1
1100	1	0010	1	2
1101	1	0011	1	3
1110	1	0100	1	4
1111	1	0101	1	5

Разработайте Verilog-описание схемы на рисунке 2, опустив элементы **DEC** и **Схема Б**. За основу возьмите разработанный в части 1 модуль, затем скомпилируйте и промоделируйте его работу, сравнивая полученные результаты с таблицей 2.

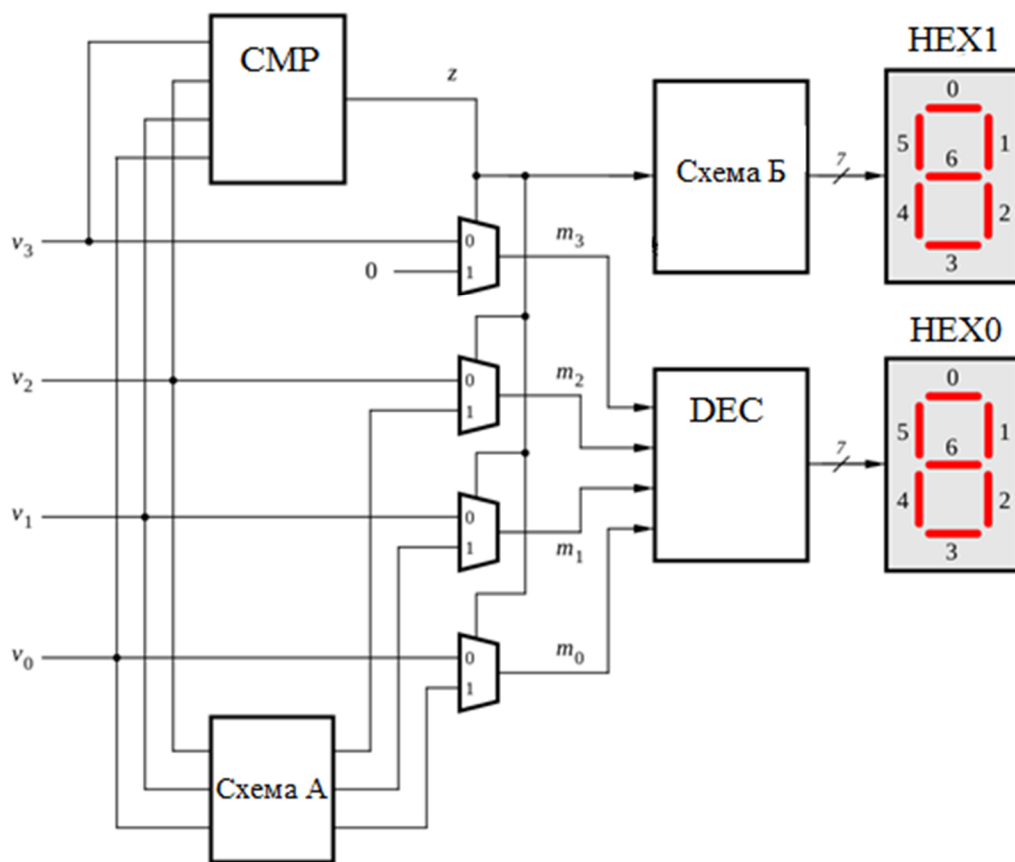


Рисунок 2 - Представление числа в двоично-десятичном формате и отображение на 7-сегментных индикаторах

Если по результатам моделирования можно сказать, что схема функционирует правильно, добавьте в проект описание элементов **DEC** и **Схема Б**.

Скомпилируйте проект, промоделируйте работу и сконфигурируйте ПЛИС как описано в лабораторной работе №1, подключив на входы $v_{3:0}$ переключатели **SW[3:0]**, а на выходы индикаторы **HEX0** и **HEX1**. Проверьте работоспособность на

отладочном стенде, изменяя положение переключателей **SW** и наблюдая значение, отображаемое на индикаторах.

Часть 3

Необходимо произвести преобразование из двоично-десятичного кода в двоичный код, и отобразить полученное значение на светодиодах **LEDG**. Например, для числа 14_{10} : $0001\ 0100_{\text{BCD}} = 1110_2 = E_{16}$.

Сложность схемы будет зависеть от количества разрядов исходного десятичного числа. В рамках данной лабораторной работы ограничимся двумя разрядами и диапазоном $0 - 99$. Необходимо две тетрады двоично-десятичного кода, представляющие десятичное число, перевести в двоичный код.

Рассмотрим следующую схему:

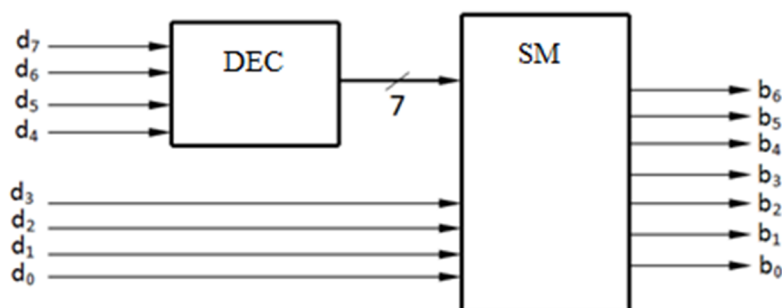


Рис.3. Преобразователь форматов двоично-десятичный в двоичный.

На схеме b_{6-0} – 7-ми разрядный двоичный результат преобразования. d_{3-0} и d_{7-4} – тетрады разряда единиц и разряда десятков исходного десятичного числа, соответственно.

DEC – дешифратор который ставит в соответствие входному значению его же, но умноженное на 10, заменяя операцию умножения. Такой подход сокращает аппаратные затраты и целесообразна из-за малого количества комбинаций входного кода второй тетрады. Ширина выходной шины дешифратора определена максимальным выходным значением дешифратора. Для реализации дешифратора используйте конструкцию **case**.

SM – 7-ми разрядный сумматор, который суммирует первую (младшую) тетраду со второй, преобразованной в дешифраторе (заменив умножение на 10).

Двоичный 7-ми разрядный результат суммы можно получить на выходе

сумматора. При увеличении разрядности десятичного числа, схема легко расширяется добавлением ещё одного дешифратора и увеличением разрядности сумматора.

Схему можно упростить:

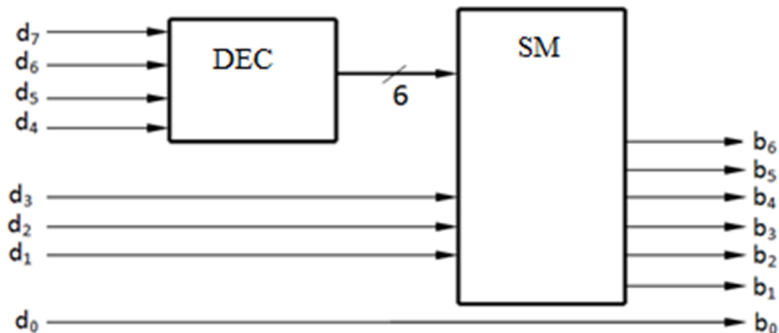


Рисунок 4 - Упрощенный преобразователь форматов двоично-десятичный в двоичный

Предлагается самостоятельно проанализировать данную схему. Обратите внимание на ширину выходной шины декодера, а также на нулевой разряд младшей тетрады – d_0 .

Реализуйте схему, изображённую на рисунке 4, в виде Verilog-описания. Тетрады десятичного числа d_{7-0} сформируйте переключателями **SW**, выходное значение отобразите на светодиодах **LEDG**.

Скомпилируйте и промоделируйте проект, а также сконфигурируйте ПЛИС как описано в лабораторной работе №1. Проверьте функциональность схемы на отладочном стенде, изменяя положение переключателей **SW** и наблюдая значение, отображаемое на светодиодах.

Индивидуальное задание.

Необходимо разработать Verilog-описание преобразователя по следующим правилам:

1. Входное значение V – двоичное 5-разрядное число в диапазоне $0_{10} - 19_{10}$ включительно. Выходное преобразованное значение M – число в двоично-десятичном коде.
2. Основание счета N преобразованного числа рассчитывается следующим образом: к числу $N=10$ прибавляется младший десятичный разряд номера вашего варианта по списку, взятый по модулю 9, если он не нулевой, иначе $N=10+1$. Получаем диапазон оснований 11-18. Младший разряд преобразованного числа занимает две позиции в записи. Например:
 $N = 15, V = 17, M = 102;$
 $N=15, V=13, M=13.$
3. Начальное значение $V=0$. Значение инкрементируется по нажатию кнопки **KEY[0]**, при этом на 7-сегментных индикаторах должно наблюдаться изменение выходного значения.

Для выполнения задания доработайте Verilog-описание схемы из части 2.

Требования к отчету.

В качестве отчета о выполнении лабораторной работы вы должны предоставить проект в САПР Quartus II, содержащий схемы, Verilog-описания, тестовые воздействия и результаты моделирования преобразователей, предложенных к разработке в методических указаниях, а также в индивидуальном задании. Продемонстрировать на учебном стенде работу преобразователя, предложенного в индивидуальном задании.

Контрольные вопросы.

1. Опишите достоинства и недостатки представления чисел в двоично-десятичном коде и поясните каждый из них.
2. Объясните, для чего в схеме на рисунке 2 используются элементы **Схема А** и **Схема Б**.

Задание для самостоятельной работы.

Необходимо самостоятельно разобраться с материалом, представленным в теоретической части, и разобраться с работой преобразователей форматов двоичный в двоично-десятичный и двоично-десятичный в двоичный во внеурочное время.

Кроме предложенных алгоритмов преобразования форматов существуют и другие. Например, алгоритм преобразования в двоично-десятичный код *Сдвиг и прибавление*. Ознакомьтесь с данным алгоритмом по предложенной ниже литературе.

Список литературы.

1. Курс лекций по дисциплине «Электротехника, электроника и схемотехника. Программируемые логические интегральные схемы»
2. Altera Development and Education Board 1. User Manual. 2006, Altera Corporation.
3. Dr. John S. Loomis, Electrical and Computer Engineering University of Dayton.
Binary to BCD Converter.
http://www.johnloomis.org/ece314/notes/devices/binary_to_BCD/bin_to_bcd.html