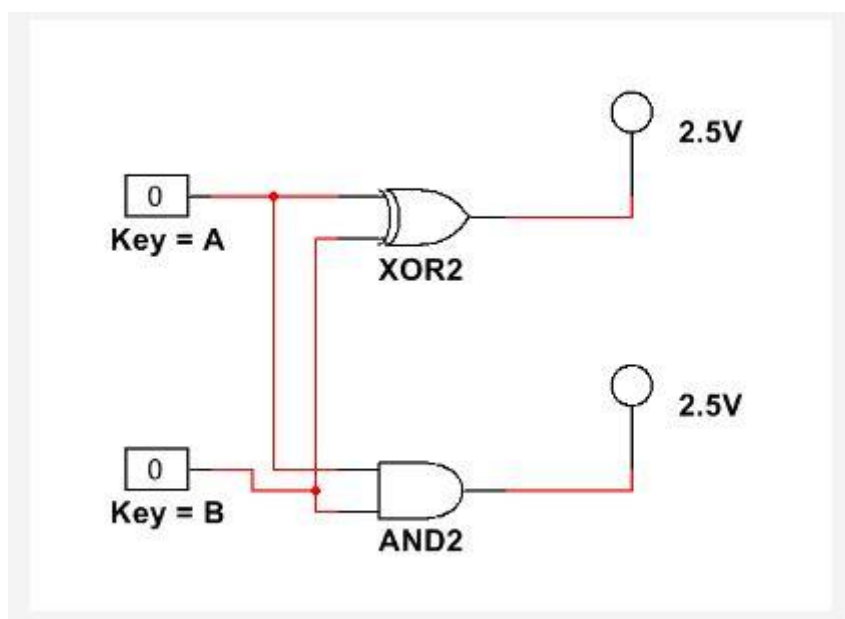


# Руководство к лабораторному практикуму:

## Введение в цифровую электронику

С использованием платы Digilent Digital Electronics для NI ELVIS III



Лабораторная работа 4:  
Преобразование двоичного кода и  
сумматоры

© 2018 National Instruments

All rights reserved. Данный ресурс и любые его части не могут быть скопированы или в любой форме воспроизведены иным способом без письменного разрешения издателя.

National Instruments относится с уважением к чужой интеллектуальной собственности и призывает к этому же своих читателей. Данный ресурс защищен законами об охране авторских прав и прав на интеллектуальную собственность. Вы имеете право передавать программное обеспечение и прочие материалы, разработанные с помощью описанного в данном ресурсе программного обеспечения, третьим лицам в соответствии с условиями приобретенной вами лицензии и другими законодательными ограничениями.

LabVIEW и National Instruments являются торговыми марками National Instruments.

Названия других упомянутых торговых марок и изделий являются собственностью их правообладателей.

**Дополнительные ограничения ответственности:** Читатель принимает все риски от использования данного ресурса и всей информации, теорий и программ, содержащихся или описанных в нем. Данный ресурс может содержать технические неточности, типографические ошибки, прочие ошибки и упущения, и устаревшую информацию. Ни автор, ни издатель не несут ответственности за любые ошибки или неточности, за обновление любой информации и за любые нарушения патентного права и прочих прав на интеллектуальную собственность.

Ни автор, ни издатель не дают никаких гарантий, включая, но не ограничиваясь, любую гарантию на достаточность ресурса и любой информации, теорий или программ, содержащихся или описанных в нем, и любую гарантию, что использование любой информации, теорий или программ, содержащихся или описанных в ресурсе, не нарушит любое патентное право или иное право на интеллектуальную собственность. РЕСУРС ПОСТАВЛЯЕТСЯ "КАК ЕСТЬ". ИЗДАТЕЛЬ ЗАЯВЛЯЕТ ОБ ОТКАЗЕ ОТ ЛЮБЫХ ГАРАНТИЙ, ЯВНО ВЫРАЖЕННЫЕ ИЛИ ПОДРАЗУМЕВАЕМЫХ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ, ЛЮБЫЕ ПОДРАЗУМЕВАЕМЫЕ ГАРАНТИИ ТОВАРНОГО СОСТОЯНИЯ, ПРИГОДНОСТИ ДЛЯ КОНКРЕТНОЙ ЦЕЛИ И НЕНАРУШЕНИЯ ПРАВ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ.

Издатель или автор не предоставляют прав или лицензий под любым патентным правом или иным правом на интеллектуальную собственность прямо, косвенно или лишением права на возражение.

НИ ПРИ КАКИХ ОБСТОЯТЕЛЬСТВАХ ИЗДАТЕЛЬ ИЛИ АВТОР НЕ НЕСУТ ОТВЕТСТВЕННОСТИ ЗА ПРЯМЫЕ, КОСВЕННЫЕ, ОСОБЫЕ, СЛУЧАЙНЫЕ, ЭКОНОМИЧЕСКИЕ ИЛИ ВТОРИЧНЫЕ УБЫТКИ, ПОНЕСЕННЫЕ ИЗ-ЗА ИСПОЛЬЗОВАНИЕ ЭТОГО РЕСУРСА ИЛИ ЛЮБОЙ ИНФОРМАЦИИ, ТЕОРИЙ ИЛИ ПРОГРАММ, СОДЕРЖАЩИХСЯ ИЛИ ОПИСАННЫХ В НЕМ, ДАЖЕ БУДУЧИ ПРЕДУПРЕЖДЕННЫ О ВОЗМОЖНОСТИ ПОДОБНЫХ УБЫТКОВ, И ДАЖЕ ЕСЛИ УБЫТКИ ВЫЗВАНЫ НЕБРЕЖНОСТЬЮ ИЗДАТЕЛЯ, АВТОРА ИЛИ ИНЫХ ЛИЦ, Применимое законодательство может не разрешить исключение или ограничение случайных или косвенных убытков, поэтому приведенные выше ограничения или исключения могут вас не касаться.

## Лабораторная работа 4: Преобразование двоичного кода и сумматоры

В первой лабораторной работе мы изучили двоичные таблицы истинности и узнали, как с их помощью разрабатывать схемы. Двухвходовые логические схемы – самые простые. В более сложных схемах используется больше комбинаций двоичных чисел. Создавать таблицы истинности для всех возможных комбинаций и перестановок – непрактично. Вместо таблиц истинности мы используем систему, преобразующую наши двоичные числа в *двоично-десятичный код (Binary-Coded Decimal – BCD)*. Каждому двоичному разряду назначается значение в зависимости от его положения в двоичном коде. Сложение этих значений даст нам двоично-десятичный код.

### Цель работы

В этой лабораторной работе студенты должны:

1. Из логических элементов спроектировать полусумматор и полный сумматор и составить их таблицы истинности
2. Соединить несколько полных сумматоров для изучения цепей переноса
3. Подтвердить правильность таблицы истинности полного сумматора.

## Необходимые инструментальные средства и технологии

Платформа: NI ELVIS III

- ✓ Руководство пользователя:  
<http://www.ni.com/en-us/support/model.ni-elvis-iii.html>
- ✓ Учебные пособия:  
[https://www.youtube.com/playlist?list=PLvcPluVaUMIWm8ziaSxv0gwtshBA2dh\\_M](https://www.youtube.com/playlist?list=PLvcPluVaUMIWm8ziaSxv0gwtshBA2dh_M)

---

Технические средства:  
плата Digilent Digital Electronics Board  
for NI ELVIS III

- ✓ Руководство пользователя  
NI Digital Electronics Board:  
<http://www.ni.com/pdf/manuals/376627b.pdf>

---

Программное обеспечение:  
NI Multisim 14.0.1 Education Version или  
выше

- ✓ Установка Multisim:  
[http://www.ni.com/gate/gb/GB\\_ACAD\\_EMICEVALMULTISIM/US](http://www.ni.com/gate/gb/GB_ACAD_EMICEVALMULTISIM/US)
- ✓ Справка Multisim Help:  
<http://www.ni.com/multisim/technical-resources/>

---

Программное обеспечение:  
NI LabVIEW FPGA Vivado 2014.4

- ✓ Установка:  
<http://www.ni.com/download/labview-fpga-module-2015-sp1/5920/en/>

**Примечание:** драйвер Digilent (Инсталлятор, указанный выше автоматически загружает на ваш компьютер инсталлятор, указанный ниже)

- ✓ Перейдите по ссылке:  
C:\NIFPGA\programs\Vivado2014\_4\data\xicom\cable\_drivers\nt64\digilent
- ✓ Установите: install\_digilent.exe

## Ожидаемые результаты

В этой лабораторной работе вы должны собрать для отчета:

- Подробные ответы на вопросы о сумматорах
- 3 таблицы истинности
- Ответы на вопросы из раздела *Заключение*

Преподавателю, скорее всего, необходимо предъявить полный отчет о работе. Узнайте у вашего преподавателя, есть ли конкретные требования к отчету или шаблон для его оформления.

## 1.1 Сведения из теории

### Binary-Coded Decimal

Value	8	4	2	1
Bit	0	0	1	1
Result	0	0	2	1

Sum = 0 + 0 + 2 + 1 = 3

Each "bit" is assigned a value

Multiply the value by the bit

Add the results together

The sum is the **binary-coded decimal** value

Рисунок 1-1 Видео Просмотр видео здесь: <https://youtu.be/YYGAPRracIY>



#### Краткое содержание видео

- Для больших схем составлять таблицы истинности непрактично, поэтому двоичный код преобразуют в двоично-десятичный.
- Полусумматоры в Multisim могут быть представлены одним компонентом, или их можно создать с помощью логических элементов AND и XOR.
- У полусумматоров два входа и два выхода
- Полные сумматоры в Multisim могут быть представлены одним компонентом, или их можно создать с помощью двух логических элементов AND и трех логических элементов XOR.
- У полных сумматоров три входа и два выхода

#### Двоично-десятичный код

Рассмотрим пример с 4-разрядным двоичным числом (четыре двоичных разряда кодируются одним десятичным числом):

0101

Самый правый разряд дает значение единиц (аналогично столбцу единиц при обычном сложении). Разряд слева от него дает значение двоек (аналогично столбцу десятков). Следующий слева разряд дает значение четверок (аналогично столбцу сотен). Значение каждого следующего слева разряда увеличивается вдвое по сравнению с предыдущим.

В нашем примере десятичное значение для указанного четырехразрядного двоичного кода равно сумме значений:

$$0 + 4 + 0 + 1 = 5$$

Если результатом преобразование является однозначное десятичное число (0-9), оно уже представлено в двоично-десятичном формате. Чтобы получить десятичное значение больше 9 из двоичного числа необходимо преобразование двоичного кода в двоично-десятичный код, например, по алгоритму Shift-Add-3 (сдвига и прибавления 3).

## Сумматоры

### Полусумматоры:

- Полусумматор выполняет двоичное сложение двух входных переменных (A и B).
- Два выхода полусумматора обозначены S (Sum – сумма) и C (Carry – перенос).
- Полусумматоры можно построить из:
  - логических элементов XOR и AND (слева).
  - компонента в Multisim (справа).

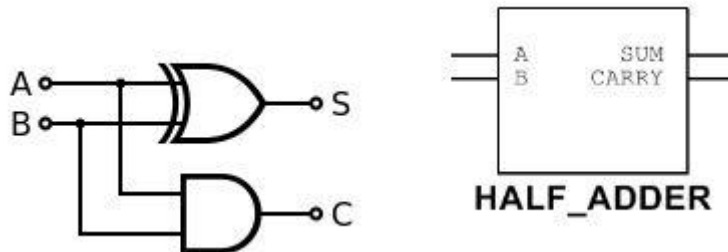


Рисунок 1-2 Полусумматор на логических элементах Рисунок1-3 Полусумматор в Multisim

### Полные сумматоры:

- Полный сумматор выполняет двоичное сложение трех переменных: A, B и  $C_{in}$  (вход переноса).
- Полные сумматоры обычно соединяются по каскадно и используются для сложения двоичных чисел с увеличивающимся количеством разрядов.
- У сумматора два выхода: суммы S и переноса  $C_{out}$ .
- В качестве полных сумматоров могут использоваться схемы на логических элементах или компонент Multisim.

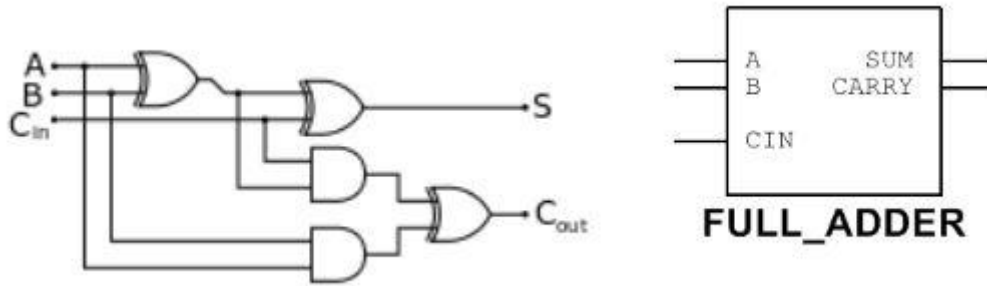


Рисунок 1-4 Полный сумматор на логических элементах Рисунок 1-5 Полный сумматор в Multisim

1-1 Почему при увеличении разрядности двоичного кода сложно использовать таблицы истинности?

---



---



---



---

1-2 Что общего у полусумматоров и полных сумматоров, чем они отличаются?

---



---



---



---

1-3 Когда сумматоры соединены друг с другом так, что выход  $C_{out}$  одного из сумматоров соединен со входом  $C_{in}$  другого, это называется распространением переноса (rippling). Как вы считаете, что это означает?

---



---



---



---



## 1.2 Моделирование схемы полусумматора

### Схемы полусумматоров

Схемы полусумматоров могут быть собраны из логических элементов.

- Запустите Multisim.
- Откройте новую схему.
  - Выберите пункт меню **File>>New**.
  - В появившемся окне выберите **Blank** и щелкните по кнопке **Create**.
- Соберите следующую схему:
  - Разместите на экране логические элементы **XOR** и **AND** из группы **Misc Digital**.
  - Добавьте две константы **INTERACTIVE\_DIGITAL\_CONSTANTS** из группы **Sources**.
  - Поместите два пробника **PROBE\_DIG\_RED** из группы **Indicators**.
  - Соедините элементы схемы, как показано на следующем рисунке:

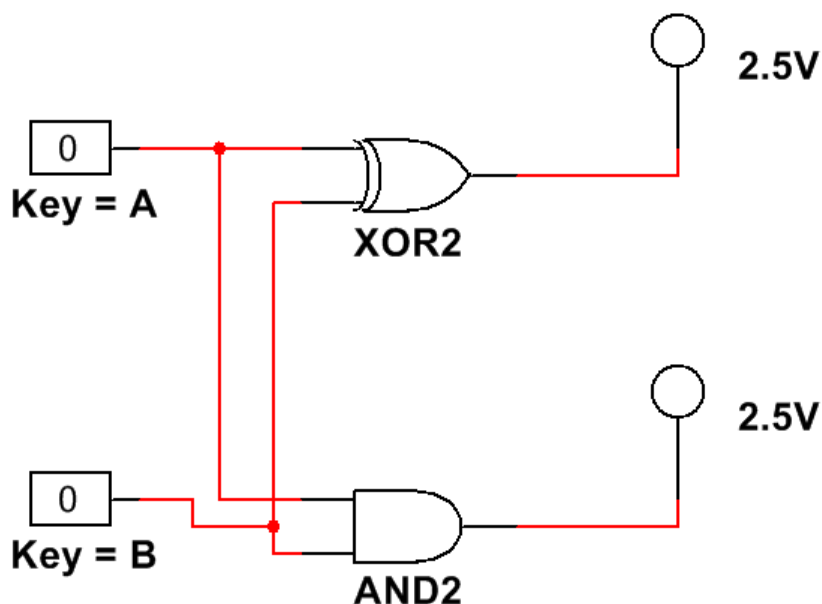


Рисунок 1-6 Схема полусумматора

- Щелкните по кнопке **Run** для запуска моделирования схемы.



Рисунок 1-7 Кнопка Run

- Используя переключатели **A** и **B**, изменяйте сигналы на входах схемы.

1-4 Заполните следующую таблицу истинности.

A	B	XOR (Сумма)	AND (Перенос)
0	0		
0	1		
1	0		
1	1		

- Остановите моделирование щелчком по кнопке **Stop**.

*Видите ли вы, как логические элементы XOR и AND представляют собой разряд суммы и разряд переноса при сложении чисел A и B?*

- $0 + 0 = 0$
- $1 + 0$  или  $0 + 1 = 1$ , переноса нет.
- $1 + 1 = 2$ , но 2 – не двоичное число. В двоичной системе счисления 2 представляется как 10. 1 – разряд переноса, а 0 – разряд суммы.

## 1.3 Моделирование схемы полного сумматора

### Схемы полного сумматора

Откройте новую схему PLD:

Соберите следующую схему:

- Разместите на экране три элемента **XOR** и два элемента **AND** из группы **Misc Digital**.
- Добавьте переключатели **SW0**, **SW1** и **SW2**.
- Добавьте светодиодные индикаторы **LED0** и **LED1**.
- Соедините элементы схемы, как показано на следующем рисунке:

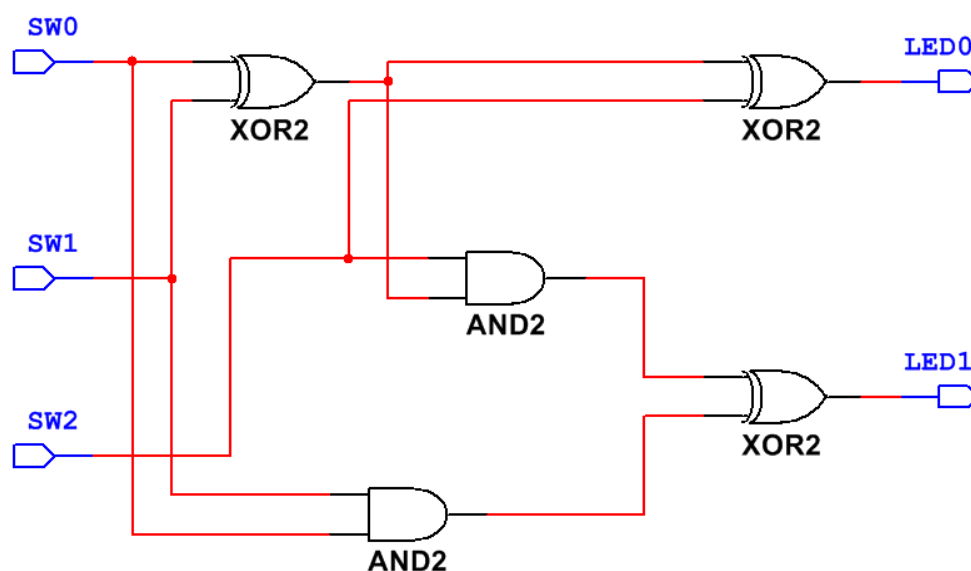


Рисунок1-9 Схема полного сумматора

- Экспортируйте схему на плату Digital Electronics. Используя переключатели на плате, изменяйте сигналы на входах схемы.

1-5 Заполните таблицу истинности.

SW0	SW1	SW2	SUM	C <sub>out</sub>
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

*Видите ли вы, как в таблице истинности представлены разряды суммы и переноса при сложении чисел A, B и C<sub>IN</sub>?*

- $0 + 0 + 0 = 0$
- $0 + 0 + 1 = 1$ , переноса нет.
- $0 + 1 + 1 = 2$ , но 2 не является двоичным числом. В двоичной системе счисления 2 представляется как 10. 1 – бит переноса, а 0 – бит суммы.
- $1 + 1 + 1 = 3$ . В двоичной системе счисления 3 представляется как 11. Первая 1 – разряд переноса, а вторая 1 – разряд суммы.

## 1.4 Разработка схемы на основе встроенных полных сумматоров

### Встроенные полные сумматоры

Откройте новую схему.

- Поместите на схему *полный сумматор*, щелкнув по кнопке **Place Misc Digital** в верхней панели.
- В появившемся окне выберите **TIL** из поля **Family** и прокрутите до пункта **FULL\_ADDER** в поле **Component**.
- Щелкните по кнопке **OK** и поместите компонент на схему слева.
- Окно временно исчезнет, а затем снова вернется уже с выбранным полным сумматором. Щелкните по кнопке **OK** и поместите следующий сумматор справа и ниже первого (оставьте между ними немного места).
- Аналогичным образом добавьте на схему еще два сумматора.
- Подключите выход **Carry** первого сумматора (самого левого) ко входу **C<sub>IN</sub>** следующего сумматора. Аналогичным образом соедините все четыре сумматора.

**Примечание:** Схема должна выглядеть следующим образом:

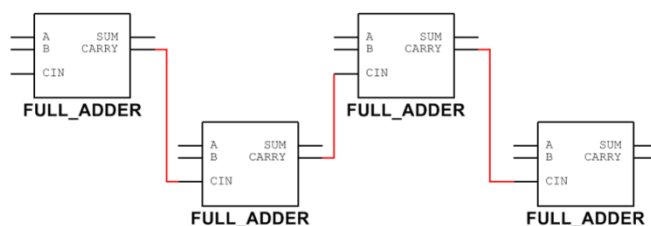


Рисунок 1-10 Схема с полными сумматорами, шаг 1

Теперь добавим источники:

- Выберите группу **Misc Digital**.
- В группе **Sources** выберите **DIGITAL\_SOURCES** и **INTERACTIVE\_DIGITAL\_CONSTANT**.
- Выберите **OK** и разместите константы на схеме.
- Разместив все источники, соедините их, как показано на рисунке:

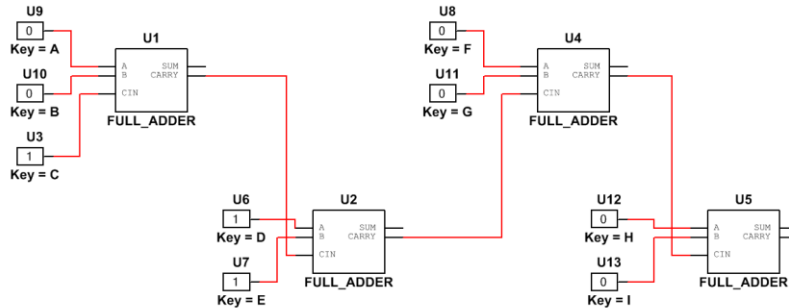


Рисунок 1-11 Схема с полными сумматорами, шаг 2

## Распространение переноса в полных сумматорах

Завершите схему, подключив для каждого сумматора пробник:

- Выберите группу **Misc Digital**.
- В группе **Indicators** выберите **PROBE** и **PROBE\_DIG\_RED**.
- Выберите **OK** и разместите пробник на схеме.
- Разместив все пробники, соедините их, как показано на рисунке:
- Измените назначение клавиш, управляющими различными входами, чтобы можно было независимо изменять значения входных переменных.

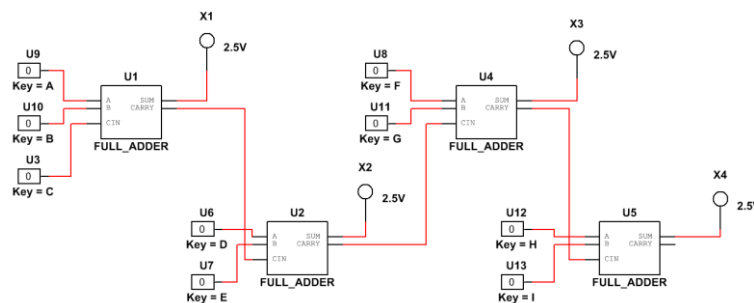


Рисунок 1-12 Схема с полными сумматорами, шаг 3

- Щелкните по кнопке **Run** для запуска моделирования схемы.



Рисунок 1-13 Кнопка Run

1-6 Изменяя значения переменных на входах первого сумматора, заполните следующую таблицу истинности.

A	B	C	SUM 1	SUM 2	SUM 3	SUM 4
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

1-7 Совпадает ли эта таблица истинности с таблицей из вопроса 1-5?

---

---

**Примечание:** Изменяйте значения переменных на входах сумматоров 2 ÷ 4 и запишите комбинацию, при которой загораются все пробники:

---

---

- Когда закончите, остановите моделирование щелчком по кнопке **Stop**.



Рисунок 1-14 Кнопка Stop

## 1.5 Заключение

1-8 При каких условиях следует использовать полусумматор? А полный сумматор?

---

---

---

---

1-9 Как используется компонент Full Adder в Multisim? Иначе говоря, почему просто не использовать элементы XOR и AND для каждого сумматора?

---

---

---

---

1-10 При преобразовании двоичных чисел в двоично-десятичный код:

- A. Разрядам сопоставляются значения в зависимости от их положения в коде
- B. Значение разряда слева в два раза больше значения соседнего разряда справа
- C. Однозначные числа уже представлены в двоично-десятичном формате
- D. Все вышеперечисленное

1-11 Полусумматоры выполняют двоичное сложение двух входных переменных с помощью двух логических элементов. Каких именно?

- A. XOR и OR
- B. XOR и AND
- C. XNOR и AND
- D. NOT и OR



1-12 Выходами полусумматора являются:

- A. Carry и  $C_{in}$
- B. Sum и Carry
- C. Sum и  $C_{in}$
- D. Carry и  $C_{out}$

1-13 Что является особенностью полных сумматоров?

- A. У них два входа
- B. У них три выхода
- C. Они передают результат суммирования (перенос) от предыдущих разрядов к следующим
- D. Все вышеперечисленное

1-14 При соединении полных сумматоров в Multisim выход Carry первого подключается ко входу \_\_\_\_\_ следующего.

- A. A
- B. B
- C.  $C_{in}$
- D. Ничего из вышеперечисленного