

# Обучение основам цифровой логики-теория, моделирование и развертывание

Обновлено 24 Мая 2019



## Обзор

Практический подход к изучению цифровой логики может быть трудным без необходимости изучения студентами сложных аппаратных описательных языков (например, языки VHDL.) Multisim Programmable логическая схема (PLD), вместе с поддержкой для ведущего Digilent уча оборудования, позволяет студентам положить основы цифровой теории в практику. Схема PLD позволяет преподавателям и студентам создавать графические логические диаграммы, подобные тем, которые находятся в учебниках, и развертывать их на Дигилентных учебных советах. В этом наборе учебных пособий мы демонстрируем, как теория цифровой логики может преподаваться с использованием учебного оборудования для обеспечения практического подхода к обучению.

Просмотр полной [обучающей серии учебников по основам цифровой логики](#)

# Развертывание цифровой логики

После того, как студенты получили представление через моделирование цифровой логики, они могут продолжить это обучение путем развертывания Digilent образовательного оборудования.

В этом учебном пособии приведен пример того, как студенты могут создать программируемый логический проект (PLD) и развернуть его на оборудовании Digilent. Учебник был написан с использованием платы Digilent Nexys 3, но это тот же процесс для всех карт Digilent.

[Начало работы с Digilent Boards в multisim tutorial](#) расскажет о процессе создания схемы PLD для используемой платы Digilent.

1. Создайте новую схему PLD для вашей платы Digilent. Подробности этого процесса можно найти в разделе [начало работы с Digilent Boards в Multisim](#) . Во время создания выберите, чтобы **снять флажок все** при выборе терминалов ввода-вывода для включения в схему.

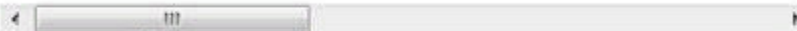


## Default operating voltages

Input connector:  VOutput connector:  VBidirectional connector:  V

## Select the defined connectors to place on the PLD:

- |                                |                               |   |                                    |
|--------------------------------|-------------------------------|---|------------------------------------|
| <input type="checkbox"/> CLK1  | <input type="checkbox"/> JB_2 | <input type="checkbox"/> JD_2             | <input type="checkbox"/> Led_6     |
| <input type="checkbox"/> HSYNC | <input type="checkbox"/> JB_3 | <input type="checkbox"/> JD_3             | <input type="checkbox"/> Led_7     |
| <input type="checkbox"/> JA_0  | <input type="checkbox"/> JC_0 | <input checked="" type="checkbox"/> Led_0 | <input type="checkbox"/> OutBlue_  |
| <input type="checkbox"/> JA_1  | <input type="checkbox"/> JC_1 | <input checked="" type="checkbox"/> Led_1 | <input type="checkbox"/> OutBlue_  |
| <input type="checkbox"/> JA_2  | <input type="checkbox"/> JC_2 | <input checked="" type="checkbox"/> Led_2 | <input type="checkbox"/> OutGreer  |
| <input type="checkbox"/> JA_3  | <input type="checkbox"/> JC_3 | <input checked="" type="checkbox"/> Led_3 | <input type="checkbox"/> OutGreer  |
| <input type="checkbox"/> JB_0  | <input type="checkbox"/> JD_0 | <input type="checkbox"/> Led_4            | <input type="checkbox"/> OutGreer  |
| <input type="checkbox"/> JB_1  | <input type="checkbox"/> JD_1 | <input type="checkbox"/> Led_5            | <input type="checkbox"/> OutRed_ ( |



Check all

Uncheck all

&lt; Back

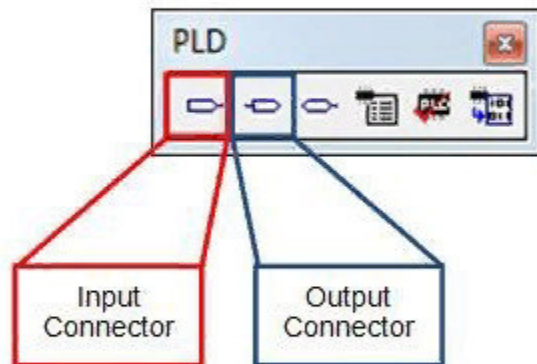
Next &gt;

Finish

Cancel

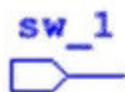
Help

2. В этом уроке мы создадим пример, который позволит студентам получить представление о функциональности логических ворот OR и Not. Чтобы добавить ввод-вывод в схему, выберите входной или выходной соединитель на панели инструментов PLD. Нажмите кнопку **входной разъем**.

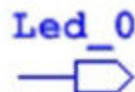


3. Это отобразит диалоговое окно "входной разъем" и предоставит список всех операций ввода-вывода на плате Digilent, назначенных в качестве входных данных. В списке выберите **переключатель 0 (sw\_0)** и нажмите кнопку **ОК**. Поместите терминал на схему.

4. Повторите процесс, чтобы добавить вход для **коммутатора 1 (sw\_1)**.



5. С помощью кнопки выходной разъем создайте выходной разъем для светодиода 0 (LED\_0).



6. Откройте базу данных компонентов (щелкните правой кнопкой мыши > разместить компонент) и поместите 2 входа и ворота.

**База данных:** Master

**группа баз данных:** PLD логическое

**семейство:** LOGIC\_GATES

**компонент:** AND2

9. Как только схема была создана, мы готовы раскрыть схему логики к доске Digilent поэтому студенты могут физически манипулировать переключатели и осмотреть реакцию на Сид. Полную информацию о развертывании карт Digilent можно найти в разделе [начало работы с платами Digilent в Multisim](#) . Выберите: **передача " экспорт в PLD** из строки меню. Здесь вы увидите три варианта экспорта. Мы хотим экспортировать в физическое оборудование, поэтому выберите **программу подключенного PLD** . нажимать далее.
10. Выберите **инструмент Xilinx**, чтобы скомпилировать проект Multisim PLD в файл программирования. Если вы установили инструменты Xilinx ISE в расположение по умолчанию, они должны автоматически заполняться. Если нет, щелкните раскрывающийся список инструмент Xilinx, выберите **вручную выбрать инструмент**, а затем перейдите в папку, в которой были установлены инструменты. Файл ограничений пользователя Xilinx содержит направления, которые сопоставляют соединители в Multisim с выводами Xilinx FPGA, номер детали **XC6SLX16**.
11. На этом этапе необходимо подключить плату Digilent к компьютеру. Чтобы проверить, выполнены ли все требования и правильно ли подключено устройство к Multisim, нажмите кнопку **Обновить**. Если плата обнаружена, отображается состояние устройства - *дата и время*, как показано ниже.

Xilinx tool:

Xilinx ISE Design Suite 14.6 32-bit

Programming file:

C:\Users\ \Desktop\ProgrammableLogicDevice1.bit

Browse...

PLD part number:

XC6SLX16

Device status: Detected - 24/09/2013 12:08:02 PM

Refresh

Advanced settings

Xilinx user constraint files (\*.ucf)

C:\Program Files\National Instruments\Circ...

Allow unmatched LOC constraints

Yes

&lt; Back

Next &gt;

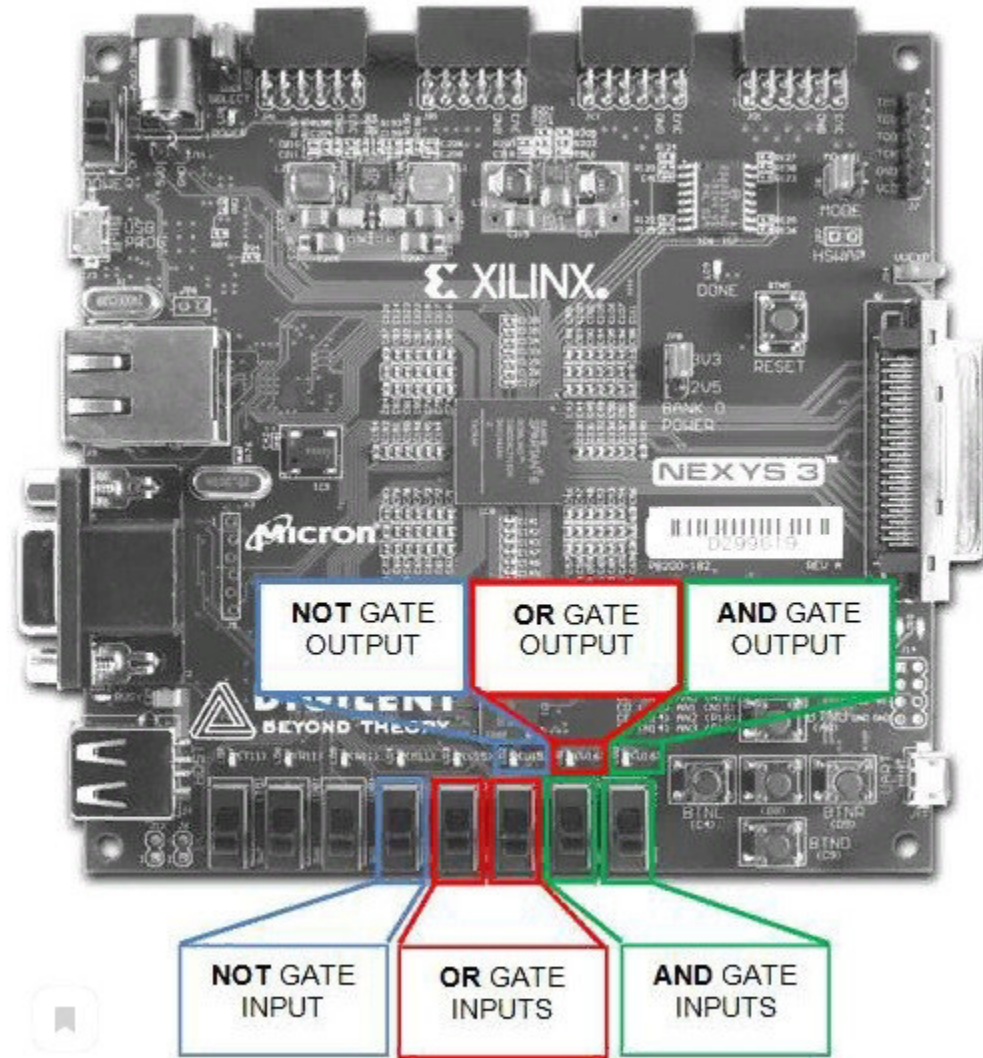
Finish

Cancel

Help

12. Чтобы продолжить, нажмите кнопку **Готово** . Начинается 11-ступенчатый процесс программирования PLD. Multisim автоматически вызывает инструменты Xilinx ISE (создает проект Xilinx, проверяет синтаксис, переводит, размещает и маршрутизирует, генерирует файл программирования и т. д.).
13. Как только код был произведен и раскрыт к доске Digilent, студент может переключить переключатели и осмотреть реакцию на интегрированном Сид.





Используя традиционные средства обучения, практический подход к изучению булевой логики невозможен до тех пор, пока студенты не пройдут более продвинутые курсы, обучающие их языкам описания оборудования (например, языки VHDL.)

Моделирование Multisim, схемы PLD и поддержка Digilent обеспечивают общее решение, позволяющее студентам учиться с помощью экспериментов.